



المدرسة الوطنية المتعددة التخصصات  
Ecole Nationale Polytechnique

Ecole Nationale Polytechnique  
Département d'Electronique

## Thèse de Doctorat en Electronique

Présenté par :

**ALKAMA Sadia**

Magister en Electronique de l'UMMTO

**Intitulé**

# Segmentation des images multi-composantes

Soutenu publiquement le **13 / 01 / 2013** devant le jury composé de :

|                     |                         |                       |                    |
|---------------------|-------------------------|-----------------------|--------------------|
| <b>Président :</b>  | HAMAMI Latifa           | Professeur            | ENP                |
| <b>Rapporteur :</b> | BERKANI Daoud           | Professeur            | ENP                |
| <b>Examineurs :</b> | BELHADJ-AISSA Aichouche | Professeur            | USTHB              |
|                     | CHIBANI Youcef          | Professeur            | USTHB              |
|                     | SMARA Youcef            | Professeur            | USTHB              |
| <b>Invité :</b>     | CHAHIR Youssef          | Maître de conférences | Université de Caen |

**ENP 2013**

## ملخص

هذه الرسالة تتناول معالجة الصور المتعددة المركبات التي تتميز أنه لا يمثل بكسل (Pixel) من قبل سلمى واحد ولكن عن طريق العديد من القيم، التي قد تكون مشابهة، أو تتوافق مع كميات فيزيائية مختلفة. تخصص مساهمات هذه الرسالة على التجزئة الشعاعية والهامشية لصور الاستشعار عن بعد و لصور ملونة. المساهمة الأولى تتعلق بإستراتيجية التجزئة الهامشية حيث تم تطوير تقنية جديدة لانصهار صور العلامات و تم اختبارها. تمت مقارنة النتائج مع تلك التي تقدم عن طريق إستراتيجية التجزئة الشعاعية عبر دالة بورسوتي (Borsotti). انها تشير الى ان هذه التقنية المقترحة توفر نتائج جيدة. كلا الإستراتيجيتان تستخدم خوارزميات التصنيف (EM)، (K-means)، (FCM)، (SOM) و (MRF). تخص المساهمة الثانية نهج MRF حيث صور العلامات اتخذ لها نموذج باستخدام طاقة جيبس (Gibbs). وتتم مقارنة سبعة أشكال من طاقة جيبس في إطار تجزئة متعددة الأطياف، واقترحت حديثا اثنين منها. استخراج وسطاء نسج جديدة للصور الملونة تشكل مساهمة أخرى من هذه الرسالة. وتستند هذه التقنية الجديدة على المشي العشوائي في الرسم البياني المستخرج من مجال التصنيف الطيفي. النتائج التجريبية لتجزئة صور ملونة منسوجة اصطناعية و حقيقية أظهرت مدى أهمية هذه الصفات.

**كلمات المفتاح:** صور متعددة المركبات، صور ملونة، صور الأقمار الصناعية، تجزئة الصور، المعالجة الشعاعية، المعالجة الهامشية، الانصهار، دالة بورسوتي (Borsotti)، وسطاء نسج، أساليب التصنيف.

## Résumé

Cette thèse porte sur le traitement des images multi-composantes, caractérisées par le fait qu'un pixel est représenté non pas par un seul scalaire, mais par plusieurs valeurs, qui peuvent être de même nature, ou bien correspondre à des grandeurs physiques différentes. Les contributions de cette thèse sont consacrées à la segmentation vectorielle et marginale des images de télédétection et des images couleur. La première contribution concerne la stratégie de segmentation marginale où une nouvelle technique de fusion des images des étiquettes est développée puis testée. Les résultats sont comparés à ceux de la segmentation vectorielle via la fonction de Borsotti. Ils montrent que technique proposée fourni de bons résultats. Dans les deux stratégies de segmentation, les algorithmes de classification Expectation Maximization (EM), K-means, Fuzzy C-means (FCM), la carte auto-organisatrice de Kohonen (SOM) et les champs aléatoires de Markov (MRF) ont été utilisés. La seconde contribution concerne l'approche MRF où le processus étiquette est modélisé en utilisant une énergie de Gibbs. Sept formes d'énergie de Gibbs, dont deux sont nouvellement proposés, sont comparées dans le cadre d'une segmentation multi-spectrale.

L'extraction de nouveaux paramètres de texture pour des images couleur texturées constitue une autre contribution de cette thèse. Cette nouvelle technique est basée sur les marches aléatoires sur graphes (random walk in graph) qui est déduite du domaine de la classification spectrale. Les résultats expérimentaux de la segmentation des images couleur texturées, synthétiques et réelles, ont montré la pertinence de ces paramètres.

**Mots-Clefs:** Images multi-composantes, images couleur, images satellitaire, segmentation d'image, traitement vectoriel, traitement marginal, fusion, fonction de Borsotti, paramètres de texture, méthodes de classification.

## Abstract

This thesis deals with multi-component image processing, characterized by the fact that a pixel is not represented by a single scalar, but by several values, which may be similar, or correspond to different physical quantities. Contributions of this thesis are devoted to vector and marginal segmentation of remote sensing and color images. The first contribution concerns the marginal segmentation strategy where a new fusion technique for labels images is developed and tested. The results are compared with those provide by vector segmentation strategy via Borsotti function. They show that the proposed technique provide good results. In the two strategies of segmentation, the clustering algorithms Expectation Maximization (EM), K-means, Fuzzy C-means (FCM), Self-Organizing Maps of Kohonen (SOM) and Markov Random Fields (MRF) are used. The second contribution deals with the MRF approach where the label process is modeled using Gibbs energy. Seven forms of Gibbs energy are compared in the framework of a multispectral segmentation, two of which are newly proposed. The extraction of new texture features for color textured images constitutes another contribution of this thesis. This new technique is based on random walks in graph which is deducted from the field of spectral classification. Experimental results on synthetic and real color textured image segmentation show the relevance of these features.

**Keywords:** multi-components images, color images, satellite images, image segmentation, vector processing, marginal processing, fusion, Borsotti function, texture parameters, classification methods.

## Remerciements

Je tiens à exprimer ma profonde gratitude à mon directeur de thèse, Monsieur Daoud BERKANI, professeur à l'Ecole Nationale Polytechnique d'Alger. Je le remercie pour la confiance qu'il m'a accordée en acceptant de me diriger dans ce travail. Cette thèse n'aurait pas pu voir le jour sans sa compréhension dans les moments difficiles, son soutien sans relâche, sa patience infaillible et ses conseils avisés, je lui en serais à jamais reconnaissante.

Que Monsieur Youssef CHAHIR, Maître de conférences à l'Université de Caen, trouve ici l'expression de ma plus profonde gratitude pour ses chaleureux accueils lors de mes différents stages scientifiques au laboratoire GREYC-ENSICAN, pour l'implication dont il a fait preuve à l'égard de mon travail, pour son aide et sa disponibilité, qui ont contribué à l'aboutissement de ce travail.

Je remercie chaleureusement les chercheurs et techniciens du Laboratoire GREYC-ENSICAN qui m'ont accueillie avec beaucoup de sympathie, une pensée spéciale au Professeur Luc Brun, qui fut le premier à m'accorder sa confiance en m'accueillant dans son équipe de recherche "Image". Je remercie tous les membres du Laboratoire pour leur gentillesse et leur bonne humeur permanente.

Je remercie Madame Latifa HAMAMI, Professeur à l'Ecole Nationale Polytechnique d'Alger, Madame Aichouche BELHADJ-AISSA, Monsieur Youcef CHIBANI et Monsieur Youcef SMARA, Professeurs à l'Université des Sciences et Technologies Houari Boumediene, pour avoir accepté de participer au jury de cette thèse.

J'adresse toute mon affection à ma famille et plus particulièrement à mon époux, à ma mère et à mes frères et sœurs. Ils embellissent ma vie, enrichissent mon existence et améliorent mon être. Leur confiance, leur patience, leur tendresse, leur amour me portent et me guident tous les jours. C'est un grand plaisir pour moi de leur dédier ce travail.

# Sommaire

|  |    |
|--|----|
| <b>Introduction générale</b> .....   | 01 |
| <b>Chapitre I : Généralités sur les images multi-composantes</b> .....                       | 05 |
| 1 Introduction .....   | 05 |
| 2 Représentation d'une image multi-composante .....  | 06 |
| 2.1 Représentation multi-scalaire .....  | 06 |
| 2.2 Représentation volumique .....   | 07 |
| 2.3 Représentation vectorielle .....   | 07 |
| 3 Types d'images multi-composantes .....   | 08 |
| 3.1 Images à composantes homogènes .....   | 08 |
| 3.2 Images à composantes quasi-homogènes .....   | 10 |
| 3.3 Images à composantes hétérogènes .....   | 12 |
| 4 Stratégies de traitement des images multi-composantes .....                                | 14 |
| 4.1 Stratégie scalaire .....   | 15 |
| 4.2 Stratégie marginale .....  | 15 |
| 4.3 Stratégie vectorielle .....  | 16 |
| 4.4 Stratégie hybride .....  | 17 |
| 5 Colorimétrie et images couleur .....   | 17 |
| 5.1 Espace RGB .....   | 18 |
| 5.2 Espace XYZ .....   | 18 |
| 5.3 Système de cône hexagonal HSV .....  | 20 |
| 5.4 Espace CIE L*a*b* .....  | 21 |
| 6 Télédétection et images MSG2 .....   | 22 |
| 6.1 Télédétection spatiale et satellite météorologique .....                                 | 22 |
| 6.2 Satellites Meteosat .....  | 23 |
| 6.3 Images MSG2 .....  | 24 |
| 7 Conclusion .....   | 26 |
| <br>   |    |
| <b>Chapitre II: Segmentation marginale et vectorielle des images multi-composantes</b> ..... | 27 |
| 1 Introduction .....   | 27 |
| 2 Stratégie de segmentation vectorielle .....  | 27 |
| 3 Stratégie de segmentation marginale et proposition d'un algorithme de fusion .....         | 28 |

|   |           |
|---|-----------|
| 3.1 Etat de l'art de la segmentation marginale .....  | 28        |
| 3.2 Méthode de fusion proposée .....  | 30        |
| 4 Segmentation des images multi-composantes en utilisant l'algorithme EM .....  | 33        |
| 4.1 Principe de l'algorithme "Expectation-Maximisation" .....   | 33        |
| 4.2 L'algorithme EM dans le cas des mélanges gaussiens .....  | 34        |
| 4.3 Stratégies d'initialisation de l'algorithme EM et proposition d'une méthode<br>d'initialisation .....   | 36        |
| 4.4 Résultats de la segmentation des images multi-composantes par l'algorithme<br>EM .....  | 37        |
| 5 Segmentation des images multi-composantes en utilisant l'algorithme K-means .....   | 39        |
| 5.1 Principe de l'algorithme K-means .....  | 39        |
| 5.2 Résultats de la segmentation des images multi-composantes en utilisant<br>l'algorithme K-means .....  | 41        |
| 6 Segmentation des images multi-composantes en utilisant l'algorithme FCM .....   | 43        |
| 6.1 Principe de l'algorithme FCM .....  | 43        |
| 6.2 Résultats de la segmentation des images multi-composantes en utilisant<br>l'algorithme FCM .....  | 45        |
| 7 Segmentation des images multi-composantes en utilisant les cartes auto-organisatrice<br>de Kohonen .....  | 47        |
| 7.1 Principe des cartes auto-organisatrices .....   | 47        |
| 7.2 Regroupement des référents .....  | 52        |
| 7.3 Choix des paramètres ajustables .....   | 53        |
| 7.4 Résultats de la segmentation des images multi-composantes par les cartes<br>auto-organisatrice de Kohonen .....                                 | 56        |
| 8 Interprétation et évaluation des résultats .....  | 58        |
| 9 Conclusion .....  | 61        |
| <br>  |           |
| <b>Chapitre III: Approche markovienne utilisant plusieurs formes d'énergies de<br/>Gibbs pour la segmentation des images de télédétection .....</b> | <b>63</b> |
| 1 Introduction .....  | 63        |
| 2 Notions d'image, de voisinage et de clique .....  | 64        |
| 2.1 Notion d'image .....  | 65        |
| 2.2 Notion de voisinage .....   | 65        |
| 2.3 Notion de cliques .....   | 65        |

|   |            |
|---|------------|
| 3 Chaîne et champs de Markov .....  | 66         |
| 4 Champs de Gibbs et Théorème de Hammersley-Clifford .....  | 67         |
| 5 Segmentation markovienne d'images multi-composantes .....   | 69         |
| 5.1 Formulation du problème de la segmentation d'images .....   | 69         |
| 5.2 Modélisation du champ des observations .....  | 70         |
| 5.3 Modélisation du champ des classes .....   | 71         |
| 5.4 Fonction d'énergie globale .....  | 73         |
| 5.5 Optimisation de la probabilité a posteriori .....   | 74         |
| 6 Les images tests .....  | 75         |
| 7 Résultats expérimentaux .....   | 76         |
| 8 Conclusion .....  | 81         |
| <br>  |            |
| <b>Chapitre IV: Segmentation markovienne d'images couleur texturées en utilisant<br/>des paramètres basés sur la classification spectrale .....</b> | <b>82</b>  |
| 1 Introduction .....  | 82         |
| 2 Etat de l'art de la segmentation d'images couleur texturées .....   | 83         |
| 3 Classification spectral .....   | 85         |
| 3.1 Formulation de la classification en utilisant les graphes .....   | 85         |
| 3.2 Notions de base sur les graphes .....   | 86         |
| 3.3 Matrices laplaciennes d'un graphe .....   | 87         |
| 3.4 Algorithmes de classification spectrale .....   | 88         |
| 4 Formalisation de la méthode de segmentation proposée .....  | 90         |
| 5 Les paramètres couleur .....  | 92         |
| 6 Les paramètres de texture .....   | 92         |
| 7 Résultats expérimentaux .....   | 95         |
| 8 Influence de l'espace de représentation couleur .....   | 99         |
| 9 Conclusion .....  | 101        |
| <b>Conclusion générale .....</b>  | <b>103</b> |
| <b>Références bibliographiques .....</b>  | <b>106</b> |

## Introduction générale

Depuis les premiers travaux dérivés du traitement du signal, le traitement et l'analyse des images numériques est devenu un domaine de recherche en pleine expansion et dont l'intérêt ne s'est jamais démenti. Les techniques de traitement des images numériques ont été tout d'abord élaborées pour les images à niveaux de gris (monochromes, ou mono-composantes) avant d'être étendues à des images multi-composantes de dimension plus élevée fournies par des systèmes de vision plus complexes mettant en œuvre un ou plusieurs capteurs. Les images numériques fournies par ces capteurs peuvent être à composantes homogènes, quasi-homogènes ou hétérogènes. Elles sont à composantes homogènes si chaque pixel de l'image est caractérisé par des composantes de même nature. On parle d'images à composantes quasi-homogène lorsque plusieurs images numériques scalaires sont produites pour caractériser une même scène selon des composantes exprimées numériquement avec la même unité de mesure. L'image est dite multi-protocoles ou à composantes hétérogènes lorsque l'information associée au pixel est obtenu par des modalités de nature différentes. Les objectifs de traitement des images multi-composantes restent sensiblement les mêmes qu'en imagerie monochrome, tels que le filtrage, la segmentation, le codage, l'interprétation..., il semble intéressant de pouvoir disposer, pour ces données multidimensionnelles, des mêmes outils que pour les images à niveaux de gris. C'est dans ce cadre que s'inscrit ce travail de thèse. Nous nous sommes en effet intéressés aux différentes approches possibles permettant de traiter ou d'analyser des images multi-composantes. Plus précisément nous nous sommes focalisés sur la segmentation qui est une étape primordiale en traitement d'images puisqu'elle conditionne la qualité de l'interprétation, puis celle de la prise de décision.

Pour segmenter une image multi-composante, trois approches sont possibles. La première est désignée par scalaire, elle consiste en la segmentation de l'image en niveaux de gris obtenue en fusionnant au préalable les différentes composantes de l'image. L'approche dite vectorielle, consiste à considérer chaque pixel comme un vecteur et à n'appliquer qu'un

seul traitement sur l'image. L'approche qualifiée de marginale, consiste à appliquer une même technique de segmentation sur chacune des composantes de l'image de manière indépendante. Cette approche marginale nécessite l'élaboration d'une technique de fusion des images des étiquettes. Peu de travaux traitent de la fusion des images des étiquettes. Cette fusion est généralement réalisée en utilisant la théorie de l'évidence. Plusieurs difficultés de cette théorie font qu'elle est difficilement exploitable. En effet, il faut par exemple réaliser au préalable une correspondance des étiquettes avant d'opérer la fusion en utilisant des fonctions de masses dont la modélisation n'est pas évidente. Notre première contribution dans cette thèse s'inscrit donc ce cadre. Nous proposons une nouvelle technique de fusion des images des étiquettes dans le cadre d'une segmentation marginale des images multi-composantes. Cette nouvelle méthode ne nécessite aucun prétraitement et est généralisée pour tous les types d'images multi-composantes. Pour rendre compte de son efficacité, plusieurs algorithmes de classification tels que "Expectation Maximization" (EM), K-means, Fuzzy C-means (FCM), les cartes auto-organisatrices de Kohonen (SOM) seront adaptés et appliqués à des images multi-composantes. Les résultats obtenus par notre nouvelle approche marginale de la segmentation seront comparés avec les résultats des segmentations vectorielles en se basant sur la fonction d'évaluation de Borsotti.

Parallèlement à la segmentation par l'approche marginale, nous nous sommes également intéressés dans cette thèse à la segmentation par l'approche vectorielle d'images multi-spectrales de télédétection. En effet, le développement de nouveaux capteurs multi-spectraux en imagerie de télédétection permet l'acquisition d'images multidimensionnelles d'une grande richesse, néanmoins, leurs interprétations se heurtent au problème de traitement du grand nombre de données multidimensionnelles. Le traitement par une approche vectorielle paraît plus adapté à l'étude de ce type d'image car celle-ci prend en considération les interactions spectrales des différentes composantes. Elle nécessite aussi moins de temps de calcul comparativement au traitement marginal surtout pour les images présentant un nombre de composantes spectrales important. La segmentation vectorielle des images de télédétection du satellite Meteosat Second Génération 2 (MSG2) par les champs aléatoires de Markov (MRF) permet la prise en compte des interactions spectrales et spatiales entre les pixels. Dans l'approche markovienne, il est nécessaire de modéliser le champ des classes (processus étiquette) en utilisant une énergie de Gibbs. Le choix d'une forme d'une fonction d'énergie peut influencer sur les résultats de la segmentation. La littérature présente plusieurs modèles de ces fonctions mais aucune étude comparative n'a été menée. C'est dans ce cadre que s'inscrit notre seconde contribution. Nous proposons deux nouvelles fonctions d'énergie et ces deux

fonctions ainsi que cinq autres usuelles sont comparées dans le cadre de la segmentation d'images MSG2.

Dans le domaine du traitement d'image, la couleur (ou la valeur radiométrique) et la texture sont plutôt considérées séparément. Cependant l'association de ces deux types de caractéristiques peut mener à des méthodes d'analyse puissantes. La texture est définie par un ensemble de paramètres qui décrivent les relations spatiales entre les pixels voisins. De nombreuses approches pour l'extraction des paramètres de texture ont été développées. Notre dernière et troisième contribution dans cette thèse consiste à introduire de nouveaux descripteurs de texture en utilisant les marches aléatoires sur graphes (random walk in graph) déduite de la classification spectrale. Rappelons que la classification spectrale est un algorithme de classification qui transforme la représentation des données initiales en une nouvelle représentation qui améliore la séparabilité des classes, de sorte que des méthodes de regroupement simples n'aient aucune difficulté à les détecter. Dans cette thèse, nous proposons d'utiliser la classification spectrale non pas en tant que technique de classification mais comme une méthode pour calculer les paramètres de texture. L'idée de base de cette méthode originale consiste à construire un graphe pondéré autour de chaque pixel de l'image. Les nœuds du graphe sont les pixels centrés dans des "patches" et les poids des arêtes reliant les nœuds sont les mesures de similitude tenant compte des distances entre patches connectés. A partir de ces mesures, une matrice de transition peut être déduite en utilisant les marches aléatoires sur graphe. Cette matrice de transition sera utilisée pour extraire les paramètres de texture. Pour chaque composante couleur, une composante texture correspondante sera calculée. Des images multi-protocoles seront créées en combinant les composantes couleurs avec les composantes de texture. Finalement ces images seront segmentées en utilisant les champs aléatoires de Markov. Pour évaluer les résultats, des images couleur texturées synthétiques et réelles sont utilisées.

La thèse est organisée en quatre chapitres de la manière suivante.

Le premier chapitre porte sur la représentation, les types et les différentes approches de traitement des images multi-composantes. Les images couleur et les images satellitaires seront développés.

Le second chapitre est consacré aux stratégies de segmentation vectorielle et marginale des images multi-composantes. La nouvelle technique de fusion des images des étiquettes que nous proposons sera détaillée. Les algorithmes Expectation Maximization (EM), K-means, Fuzzy C-means (FCM) et les cartes auto-organisatrices de Kohonen seront utilisés sur des images couleur et des images MSG2 pour comparer les résultats de la segmentation marginale

proposée avec ceux de la segmentation vectorielle. L'évaluation sera réalisée en utilisant la fonction de Borsotti.

Le troisième chapitre est consacré à la segmentation vectorielle des images MSG2 en utilisant les champs aléatoires de Markov. Dans cette approche, le champ des classes (processus étiquette) est modélisé par une énergie de Gibbs. Sept formes d'énergie de Gibbs, sont testées dans ce chapitre, dont deux sont nouvellement proposés. Les résultats seront aussi évalués en utilisant la fonction de Borsotti.

Le dernier chapitre traite de la segmentation des images couleur texturées. De nouveaux paramètres de textures sont extraits en utilisant les marches aléatoires sur graphes (random walk in graph) qui est déduite du domaine de la classification spectrale. Un rappel sur la classification spectrale est donné avant de développer la technique utilisée pour l'extraction des paramètres de texture. Pour se rendre compte de l'efficacité de ces paramètres, nous réalisons la segmentation markovienne de nombreuses images couleur texturées synthétiques et réelles.

Nous terminons par une conclusion générale et quelques perspectives à ce travail.

# Chapitre I

## Généralités sur les images multi-composantes

### 1 Introduction

L'analyse des images numériques est apparue dans les années 60, avec déjà des travaux sur l'élimination du bruit et la reconnaissance de formes [1, 2, 3]. Depuis, ce domaine a été abondamment étudié, touchant des secteurs très différents (industriel, médical, militaire, artistique,...) avec des objectifs très variés (mesure, compréhension, compression pour la transmission ou le stockage, ...).

Initialement, l'image numérique a été définie comme étant une représentation bidimensionnelle d'une scène tri-dimensionnelle et où à chaque élément de base de cette image ou pixel est associée une grandeur numérique scalaire. Ceci conduit à représenter une image numérique par la fonction  $Y$  suivante :

$$\begin{aligned} Y: \mathbb{N}^2 &\rightarrow \mathbb{N} \\ (i,j) &\rightarrow Y(i,j) \end{aligned} \tag{I.1}$$

où  $i$  et  $j$  représentent les coordonnées discrètes du pixel dans le plan et  $Y(i,j)$  la grandeur attachée à ce pixel, bien souvent l'intensité lumineuse, généralement limitée à un codage compact sur 8 bits ( $0 < Y(i,j) < 255$ ).

L'analyse et le traitement des images ont été développés initialement pour ces images scalaires (niveaux de gris), à l'époque où les systèmes d'acquisition et de calcul ne permettaient pas d'envisager des applications prenant en compte des structures plus complexes. L'évolution de la technologie des capteurs d'images et l'accroissement de la puissance de calcul et de stockage des systèmes informatiques a permis le développement du traitement des images dont la structure est plus complexe que celle des images scalaires et qui sont dénommées par le terme générique d'images multi-composantes. La représentation proposée pour les images scalaires (Eq. I.1) doit être étendue pour tenir compte des informations supplémentaires attachées à chaque pixel. Les principales représentations envisageables sont multi-scalaire, volumique et vectorielle.

Les images multi-composantes peuvent être classées, selon la nature des composantes qui la constituent en trois catégories principales :

- les images à composantes homogènes (multi-temporelles et polarimétriques),
- les images à composantes quasi-homogènes (multi-spectrales) et
- les images à composantes hétérogènes (multi-protocoles).

## 2 Représentation d'une image multi-composante

Les trois types de représentation des images multi-composantes sont "multi-scalaire", "volumique" et "vectorielle" [4]. Le choix entre ces différentes représentations dépend essentiellement de la nature de l'image. Mais, dans la mesure où une image peut accepter plusieurs représentations, ce choix pourra parfois être lié à la stratégie de traitement envisagé.

### 2.1 Représentation multi-scalaire

Dans ce cas de figure, il s'agit de représenter une image à  $D$  composantes par une série de  $D$  plans scalaires (monochromes) de résolution  $L \times M$  ( $L$  étant le nombre de lignes dans l'image et  $M$  le nombre de colonnes). La représentation monochrome (Eq. I.1) est alors modifiée pour donner la fonction  $Y$  d'une image numérique multi-composante de la manière suivante :

$$Y: \mathbb{N}^2 \rightarrow \mathbb{N}^D$$

$$(i,j) \rightarrow \{Y^1(i,j), Y^2(i,j), \dots, Y^D(i,j)\} \quad (\text{I.2})$$

La figure ci-dessous illustre graphiquement ce type de représentation.

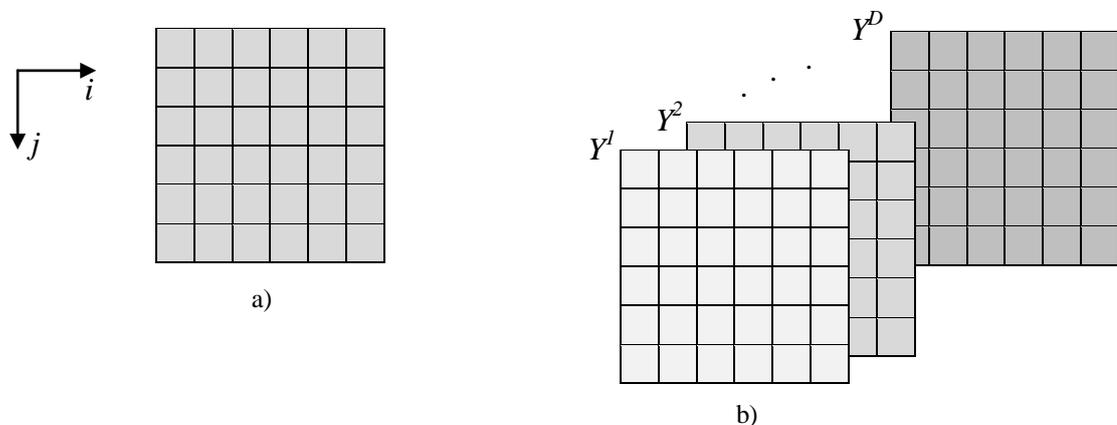


Fig. I.1 : a) Représentation d'une image scalaire  $6 \times 6$   
b) Représentation multi-scalaire d'une image multi-composante  $6 \times 6 \times D$

## 2.2 Représentation volumique

Cette représentation consiste à ajouter aux deux variables  $i$  et  $j$  de la représentation scalaire (Eq. I.1) une troisième variable apportant le caractère multi-composante. Si  $z$  désigne cette variable supplémentaire, la représentation volumique de l'image multi-composante est alors définie par :

$$\begin{aligned} Y: \mathbb{N}^3 &\rightarrow \mathbb{N} \\ (i,j,z) &\rightarrow Y(i,j,z) \end{aligned} \quad (\text{I.3})$$

A chaque cellule élémentaire est associée une grandeur scalaire. Cette représentation est utilisée lorsque la variable supplémentaire  $z$  peut prendre un grand nombre de valeurs. Une illustration graphique de cette représentation est donnée sur la figure ci-dessous.

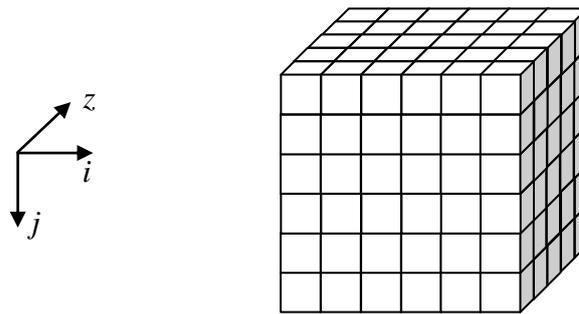


Fig. I.2 : Représentation volumique dans le cas d'une image multi-composante 6x6x5

Il est à noter que dans cette représentation un ordre implicite sur l'axe  $z$  complémentaire est établi. Dans le cas où  $z$  est une variable temporelle dans les images multi-temporelles, l'ordre chronologique est évident et dans le cas où elle est une variable fréquentielle pour les images multi-spectrales, l'ordre des fréquences croissantes semble approprié. Cet ordre implicite de l'axe complémentaire limite sa généralisation pour toutes les images multi-composantes. En effet il ne semble pas envisageable, par exemple, d'établir un ordre des modalités dans le cas des images multi-modalités.

## 2.3 Représentation vectorielle

Toutes les images multi-composantes peuvent être représentées de manière vectorielle. A chaque pixel est alors associé un vecteur contenant les différentes grandeurs de ce pixel pour les différentes valeurs du paramètre complémentaire. Si  $D$  est le nombre total de composantes dans l'image multi-composante, la représentation vectorielle prend alors la forme :

$$Y: \mathbb{N}^2 \rightarrow \mathbb{N}^D$$

$$(i,j) \rightarrow \vec{Y}(i,j) = \begin{bmatrix} Y^1(i,j) \\ Y^2(i,j) \\ \vdots \\ Y^D(i,j) \end{bmatrix} \quad (\text{I.4})$$

Une illustration graphique de ce type de représentation est donnée sur la figure ci-dessous. C'est une représentation plane ou à chaque pixel est associé un vecteur.

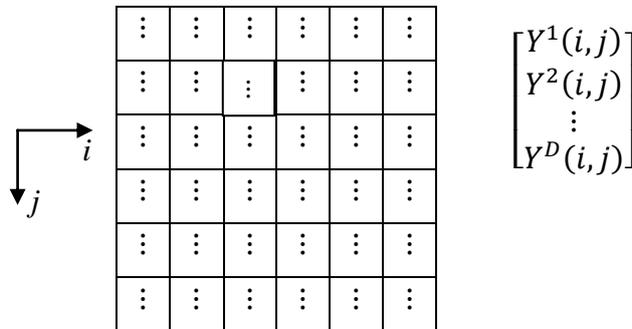


Fig. I.3 : Représentation vectorielle d'une image 6x6

Dans ce type de représentation, il est aussi nécessaire d'établir un ordre entre les différentes composantes de l'image sauf que cet ordre n'est pas contraignant comme pour la représentation volumique.

### 3 Types d'images multi-composantes

Selon la nature des composantes, les images multi-composantes sont classées en trois catégories principales: les images à composantes homogènes, à composantes quasi-homogènes et à composantes hétérogènes [4,5].

#### 3.1 Images à composantes homogènes (multi-dates et polarimétriques)

Les images sont dites à composantes homogènes si chaque pixel de cette image est caractérisé par des composantes de même nature, c'est-à-dire, représentant la même grandeur physique. L'image admet les trois représentations précédentes mais plus particulièrement les représentations multi-scalaires en une série de  $D$  plans scalaires de résolution  $L \times M$ , ou volumique de résolution  $L \times M \times D$ . La nature de la troisième dimension, peut être, selon l'application, temporelle ou polarimétrique.

Les images multi-temporelles ou multi-dates sont des images de la même zone acquises à des "dates" différentes formant ainsi des séquences d'images. Les séquences d'images

permettent par exemple d'accéder aux informations de mouvement des objets se déplaçant dans la scène. Ils permettent aussi d'étudier l'évolution des structures d'une date à l'autre, ou d'exploiter la redondance des informations recherchées pour conférer aux traitements une meilleure robustesse. La figure I.4 présente une série de 9 images fournies par le satellite MSG2 (Meteosat Second Génération 2) de la zone ouest Africaine.

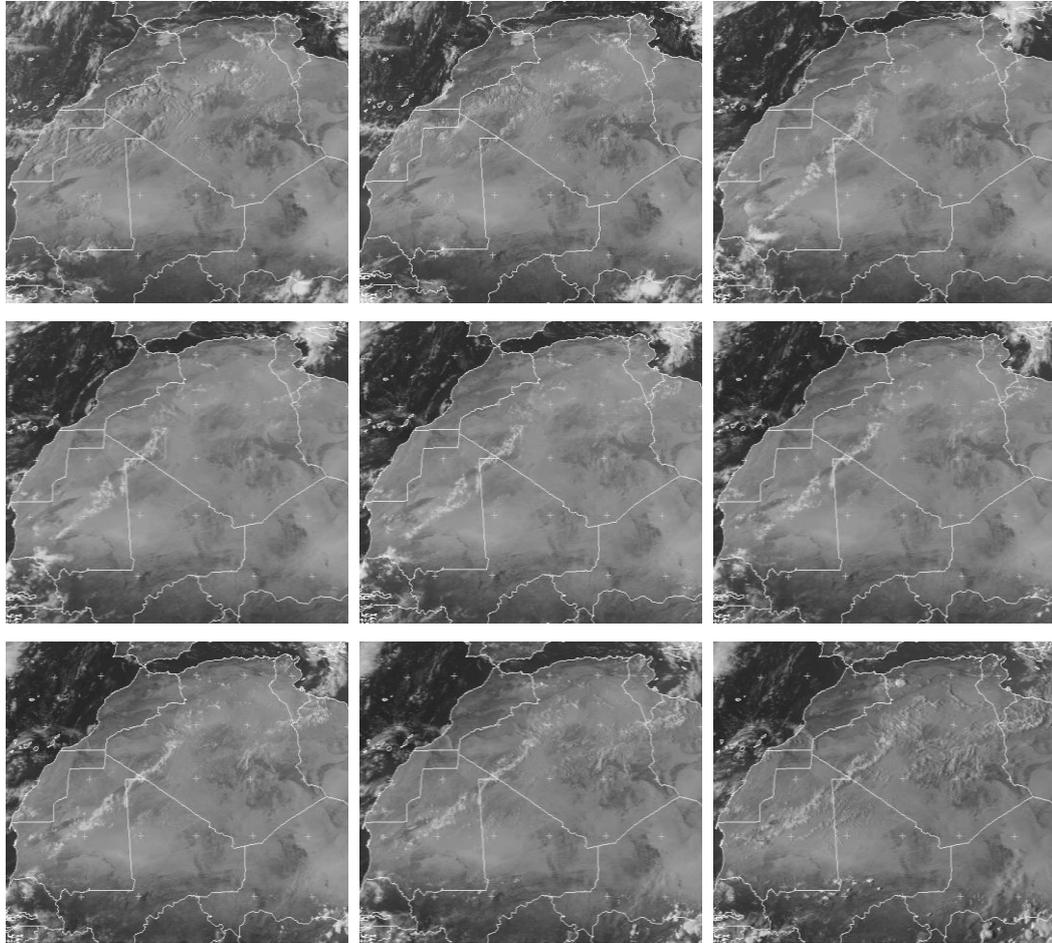


Fig. I.4 : Image multi-temporelle dans le domaine visible "VIS0.6" du satellite MSG2 prise le 25/09/2011 entre 8H et 16H à une heure d'intervalle [6]

Les images polarimétriques sont des images fournies par les radars à synthèse d'ouverture (RSO). Elles sont le résultat de la rétrodiffusion d'une onde électromagnétique. La polarisation de l'onde électromagnétique à l'émission et à la réception permet d'obtenir jusqu'à quatre polarisations différentes, généralement dénotées HH, HV, VH et VV, H pour des polarisations horizontales et V pour des polarisations verticales. La polarimétrie radar (PolSAR) exploite la phase de l'onde reçue mesurée dans différents couples de polarisations émission-réception pour en déduire des informations sur les mécanismes de rétrodiffusion. Les données dans différents couples de polarisations émission-réception, sont combinées dans des plans de l'image, constituant ainsi des images multi-composantes polarimétriques. La

figure I.5 présente une image polarimétrique du radar RADARSAT-2 suivant les polarisations HH et HV du nord canadien [7] et la figure I.6 représente une composition colorée de l'image polarimétrique de E-SAR, de la région d'Oberpfaffenhofen [8].

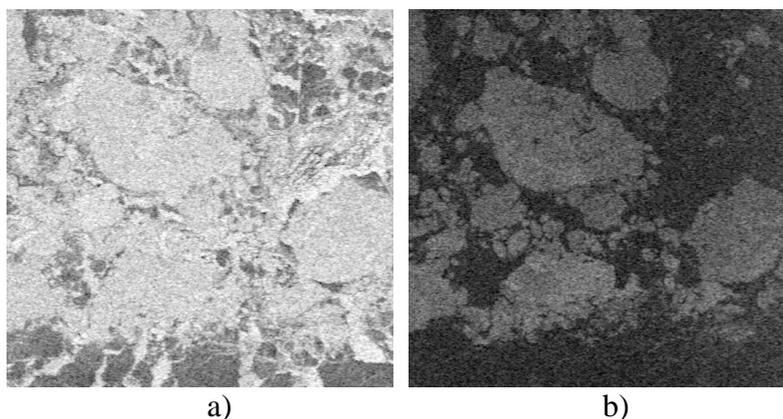


Fig. I.5 : Image RADARSAT2 polarimétrique représentant une mer de glace canadienne suivant la polarisation a) HH et b) HV



Fig. I.6 : Composition colorée de l'image de la bande L de E-SAR, de la région d'Oberpfaffenhofen, (R=HH-VV, G=2HV, B=HH+VV) [8]

Les RSO fournissent aussi des séries temporelles d'images d'amplitude. L'amplitude de l'onde reçue mesure le coefficient de rétrodiffusion du sol pour le couple de polarisation émission-réception utilisé. Après recalage et calibration, les séries temporelles d'images d'amplitude mono-polarisation constituent des images multi-composantes homogènes, porteuses d'informations sur les évolutions de l'occupation du sol.

### 3.2 Images à composantes quasi-homogènes (multi-spectrales)

Les images sont dites à composantes quasi-homogènes si chaque pixel de cette image est caractérisé par des composantes exprimées numériquement avec la même unité de mesure, par exemple des énergies dans des bandes de fréquences différentes. La représentation

rigoureuse pour ce type d'images est la représentation vectorielle à cause du fait qu'un pixel désigne une entité formellement indivisible. La prise en compte de chaque composante comme une entité distincte n'est pas appropriée bien qu'elle soit mathématiquement envisageable.

Parmi ces images, il y a bien sûr les images couleur, avec leur habituelle représentation dans l'espace colorimétrique primaire RVB (Rouge, Vert et Bleu) ou en anglais RGB (Red, Green and Blue). Ce codage s'appuie sur les expériences de Young [9], validées ultérieurement par d'autres études [10, 11], selon lesquelles toute couleur peut être synthétisée par l'addition de trois stimuli primaires de couleurs en proportions appropriées. Un exemple d'image couleur avec ses trois composantes RGB est représenté sur la figure I.7.

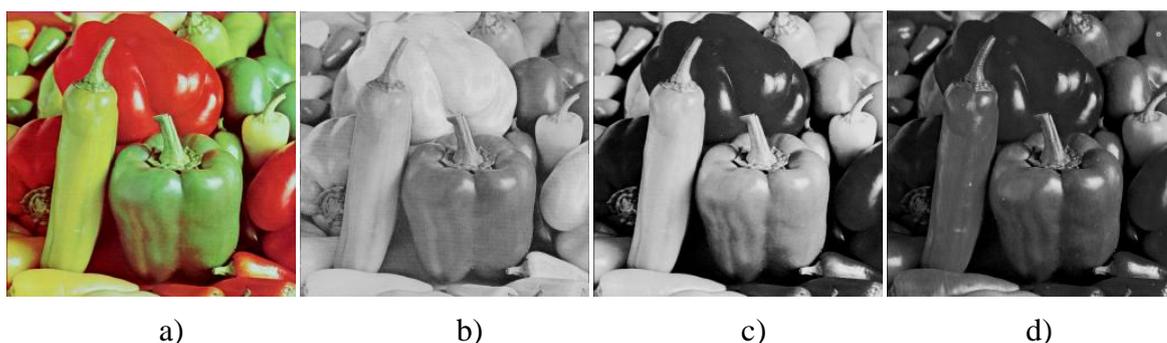


Fig. I.7 : a) L'image couleur Peppers et ses trois composantes b) Rouge, c) Verte et d) Bleu

Un autre exemple est celui des images multi-spectrales dans lesquelles les scènes sont observées à travers des filtres répartis sur différentes bandes donnant autant d'images mono-composantes. L'exemple typique de ces images sont les images satellites, présentant plusieurs bandes spectrales (trois pour SPOT4 et SPOT5, six pour LANDSAT7 et douze pour MSG2 par exemple). Lorsque le nombre de bandes devient très important, on parle d'images hyper-spectrales, obtenues par acquisition à travers un banc de filtres à bandes très étroites. Par exemple, le capteur CASI [12] fournit des images aériennes ayant plusieurs dizaines de composantes, et les images AVIRIS (*Airborne Visible InfraRed Imaging Spectrometer*) [13] peuvent comporter jusqu'à 224 bandes spectrales. A titre d'illustration, la figure 1.8 présente les composantes d'une image MSG2 de la région nord ouest africaine.

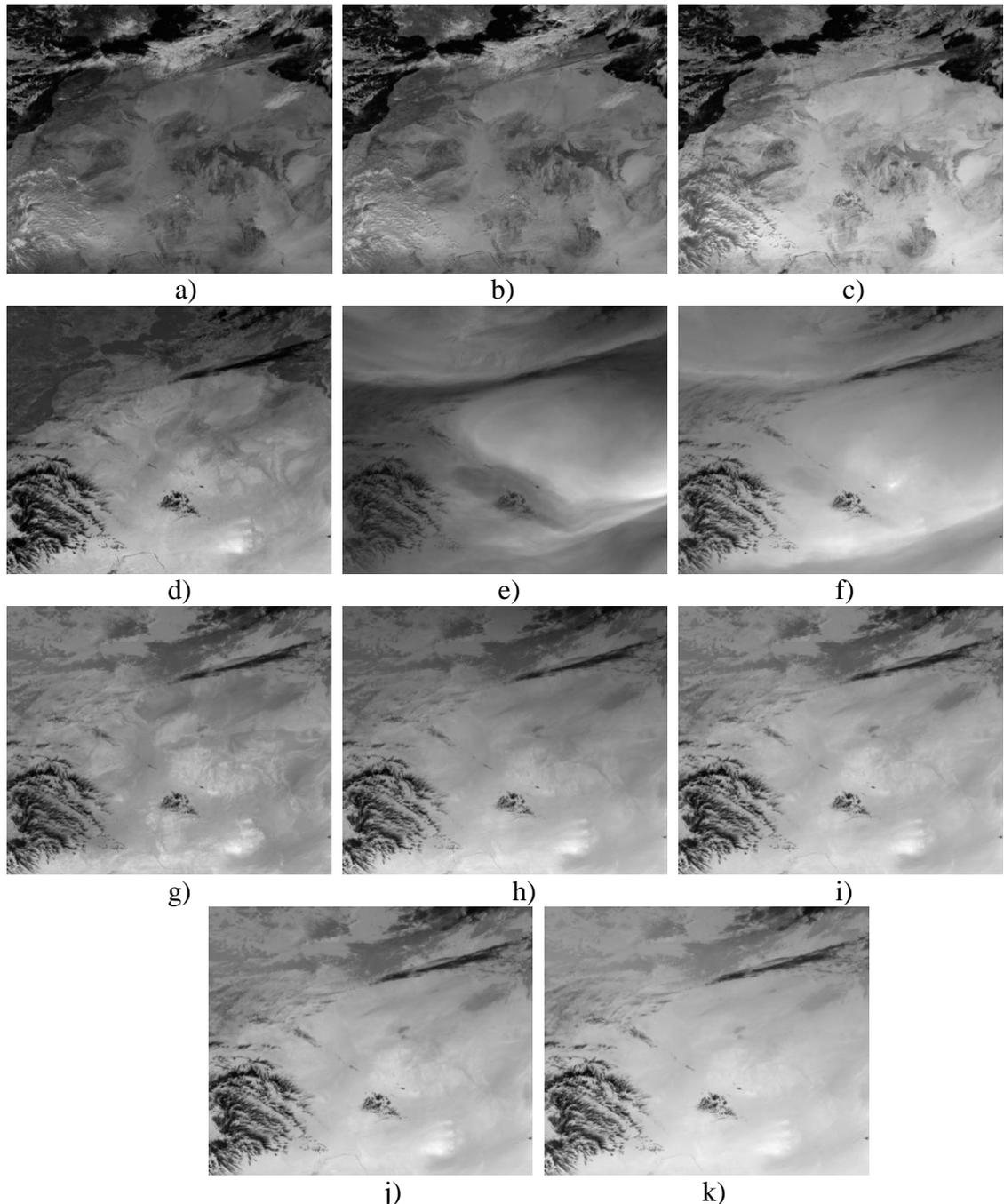


Fig. 1.8 : Les composantes de l'image MSG2 du 01/10/2006 à 12h de la région nord ouest africaine

a) canal VIS0.6 de bande  $[0.56-0.71] \mu\text{m}$  b) VIS0.8 de bande  $[0.74-0.88] \mu\text{m}$  c) NIR1.6 de bande  $[1.5-1.78] \mu\text{m}$  d) NIR3.9 de bande  $[3.48-4.36] \mu\text{m}$  e) WV6.2 de bande  $[5.35-7.15] \mu\text{m}$  f) WV7.3 de bande  $[6.85-7.85] \mu\text{m}$  g) IR8.7 de bande  $[8.3-9.1] \mu\text{m}$  h) IR9.7 de bande  $[9.38-9.94] \mu\text{m}$  i) IR10.8 de bande  $[9.8-11.8] \mu\text{m}$  j) IR12.0 de bande  $[11-13] \mu\text{m}$  k) IR13.4 de bande  $[12.4-14.4] \mu\text{m}$

### 3.3 Images à composantes hétérogènes (multi-protocoles)

Dans les images à composantes hétérogènes, les composantes d'un pixel ne sont pas de même nature. Les différents plans de l'image multi-composante sont obtenus par l'utilisation de sources de natures différentes. De ce fait, ni la représentation volumique, ni la représentation vectorielle ne peuvent être directement envisagées, seule la représentation

multi-scalaire est adaptée. Le mode de représentation vectoriel sera adapté à ces images, à condition qu'elles soient recalées et homogénéisées.

Un exemple d'image multi-composante à composantes hétérogènes est la représentation des images couleur dans des espaces perceptuels. Parmi les modèles perceptuels les plus utilisés en pratique, nous mentionnons HSL (Teinte (Hue), Saturation, Luminosité), HSI (Teinte, Saturation, Intensité) et HSV (Teinte, Saturation, Valeur). À part les différences données par le calcul des trois composantes à partir des composantes primaires RGB, les trois modèles sont basés sur la même représentation d'une couleur : ils ont une composante de teinte H qui donne la nuance de la couleur, une composante de saturation S qui donne la pureté (qui est une mesure de la quantité de gris qui est mélangée avec la couleur pure) et une composante d'intensité (L, I ou V) qui donne la luminosité de la couleur. La figure I.9 fournit un exemple d'image couleur dans l'espace colorimétrique HSV.

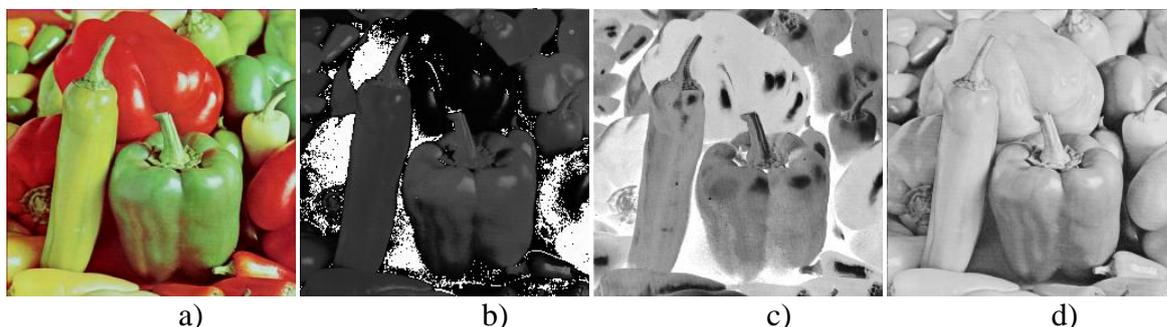


Fig. I.9 : a) L'image couleur Peppers et ses trois composantes b) H, c) S et d) V

Les images à composantes hétérogènes peuvent aussi être créées par l'utilisation de sources de natures différentes. Par exemple, une image traditionnelle de luminance peut être combinée avec une image de profondeur. Pour cette dernière, la grandeur scalaire associée à chaque pixel est proportionnelle à la distance entre l'objet et le capteur (voir Fig. I.10).

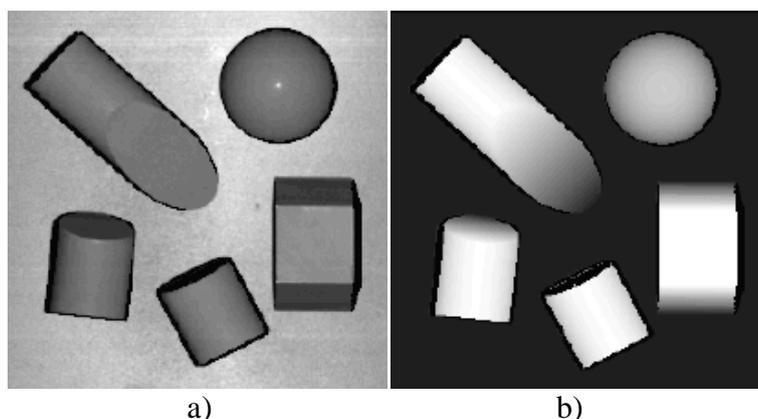


Fig. I.10 : Image multi-protocole, associant a) luminance et b) profondeur [4]

En imagerie médicale, les images multi-protocoles sont aussi fréquemment utilisées. La figure I.11 présente deux images IRM acquises en protocole "dual spin echo".

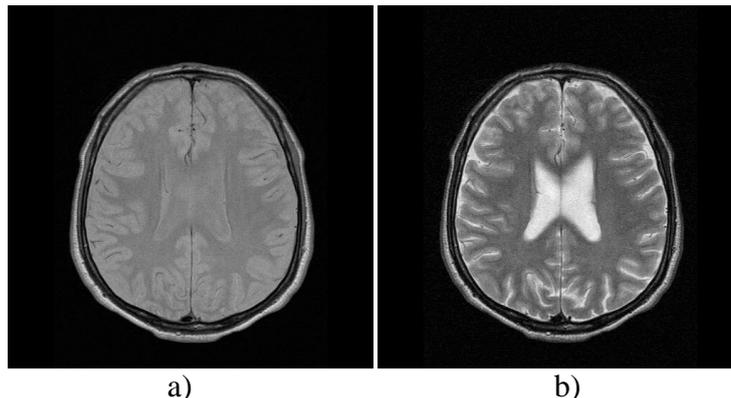


Fig. I.11 : Image IRM multi-protocole [4]

- a) Image pondérée en T2, temps d'écho=85ms, temps de répétition=3000ms  
 b) Image pondérée en DP (densité de protons), temps d'écho=14ms, temps de répétition=3000ms

On peut aussi créer des images multi-protocoles à partir de l'analyse, par exemple, d'une seule image monochrome. Ainsi Gabbouj [14] attache à chaque pixel plusieurs composantes constituées par des moyennes locales calculées sur des masques de tailles différentes. De même, Bolon [15] propose de définir pour chaque pixel d'une image monochrome, le vecteur constitué par les résultats de différentes segmentations. Nous proposons dans le chapitre IV, de créer des images multi-protocoles en attachant à chaque pixel des composantes couleurs et des composantes de texture [16].

#### 4 Stratégies de traitement des images multi-composantes

Selon la nature des composantes de l'image, certaines représentations peuvent ne pas avoir de sens. Par exemple, une image profondeur/luminance peut être représentée de manière multi-scalaire ou, à la rigueur, vectorielle, mais ne peut s'envisager dans une représentation volumique. Selon la nature des images, ces représentations peuvent être assez proches, et le passage de l'une à l'autre se fait alors aisément. Ainsi, dans le cas d'une image couleur où chaque pixel est caractérisé par un vecteur constitué des trois composantes R, G et B, la représentation volumique s'obtient en envisageant une fonction de 3 variables où la troisième variable ne peut prendre que trois valeurs correspondant aux fréquences de R, de G et de B, et qui se représenterait graphiquement par un volume de taille  $L \times M \times 3$ . L'ordre sur le troisième axe serait alors celui des fréquences.

Dans tous les cas, le choix d'une représentation est guidé par la stratégie de traitement envisagée. Pour traiter les images multi-composantes, trois stratégies peuvent être adoptées: il

s'agit des stratégies scalaires, marginales ou vectorielles [17]. Contrairement à la troisième, les deux premières ne prennent pas en compte l'aspect vectoriel des images multi-composantes ; en particulier, elles ignorent les corrélations qui peuvent exister entre les différentes composantes. En contrepartie, elles permettent de bénéficier de la nombreuse panoplie des traitements déjà disponibles pour les images scalaires. Une quatrième stratégie, dite hybride, peut être envisagée : elle permet de réintroduire en partie de l'aspect vectoriel dans les stratégies marginales.

#### 4.1 Stratégie scalaire

L'approche scalaire du traitement d'images multi-composantes consiste à fusionner tout d'abord les différentes composantes en une seule puis à traiter l'image en niveaux de gris ainsi obtenue. Le passage d'une image multi-composante en une image monochrome peut se faire par exemple par la transformée de Karhunen-Loeve ou par une quantification. On se ramène alors à un problème de traitement scalaire ou tout l'arsenal des méthodes de traitement des images monochromes peut être utilisé. Cette stratégie est illustrée à la figure I.12.

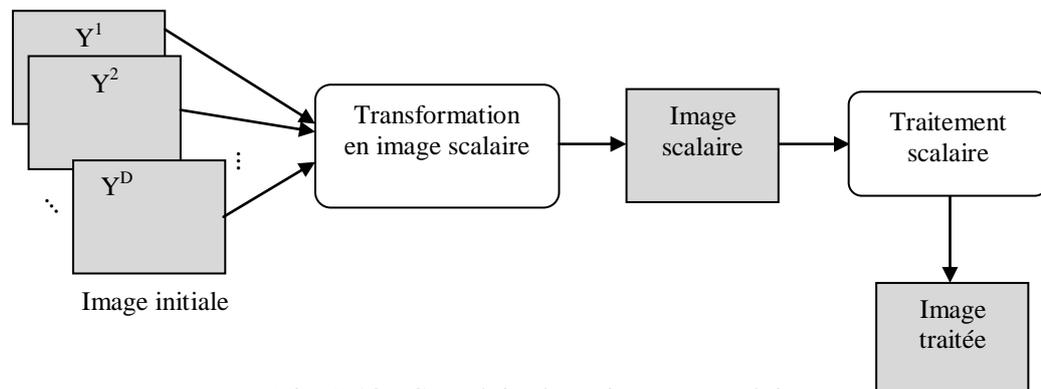


Fig. I.12 : Stratégie de traitement scalaire

#### 4.2 Stratégie marginale

Cette stratégie s'appuie sur la représentation multi-scalaire. Elle consiste à traiter chaque composante séparément en utilisant des traitements mono-composantes, empruntés à l'imagerie monochrome (Fig. I.13). Toutes les méthodes définies en imagerie monochrome sont alors directement exploitables, sans aucune adaptation. Cette stratégie demande autant de traitements qu'il y a de composantes, ce qui peut se révéler coûteux en temps de calcul lorsque le nombre de composantes est élevé. L'intérêt de cette approche est qu'elle autorise l'utilisation de traitements scalaires différents pour chaque composante. De par sa structure, cette approche ignore totalement la dépendance pouvant exister entre les différentes

composantes, délaissant ainsi une information pouvant participer à l'amélioration des performances des traitements. Cette approche présente également des limites quant à la nature des traitements envisageables. Par exemple, une opération de segmentation ne peut se faire selon le schéma de la figure I.13. En effet, il serait nécessaire, dans ce cas, d'ajouter une autre étape dans le traitement qui consisterait à fusionner les plans traités en un seul résultat.

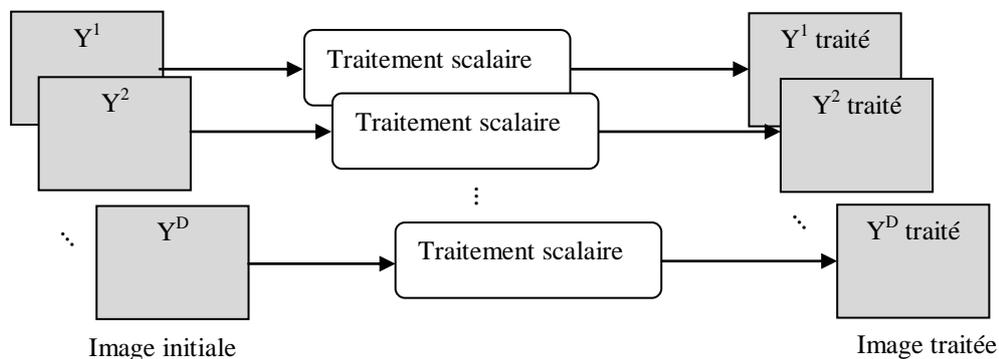


Fig. I.13 : Stratégie de traitement marginale

### 4.3 Stratégie vectorielle

Contrairement à l'approche marginale, l'approche vectorielle traite de manière globale l'ensemble des composantes, le pixel est alors considéré comme un vecteur à plusieurs composantes et le traitement s'opère sur ces vecteurs. Cette stratégie est utilisée avec les représentations volumique ou vectorielle. La figure I.14 illustre celle-ci en utilisant la représentation multi-scalaire, de manière à bien faire ressortir la différence par rapport à l'approche marginale.

L'intérêt de cette approche est la prise en compte du contexte multi-composante en plus du fait qu'elle ne nécessite qu'un seul traitement quelque soit le nombre de composantes de l'image. Ce dernier avantage doit être modéré par une complexité accrue de ce seul traitement, complexité directement dépendante du nombre de composantes et des interactions entre composantes à l'intérieur des traitements.

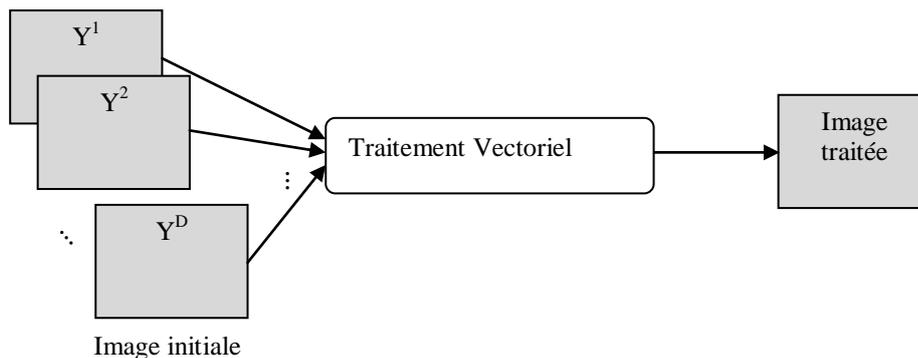


Fig. I.14 : Stratégie de traitement vectorielle

#### 4.4 Stratégie hybride

Cette stratégie est une hybridation des stratégies marginale et vectorielle. C'est une solution intermédiaire qui consiste à garder la structure marginale et à définir les traitements en tenant compte de l'ensemble des composantes (voir Fig. I.15).

Elle est typiquement utilisée pour les traitements adaptatifs. L'analyse locale, au voisinage du pixel courant de l'ensemble des composantes permet de décider de la nature des traitements qui vont ensuite être appliqués à ce pixel composante par composante.

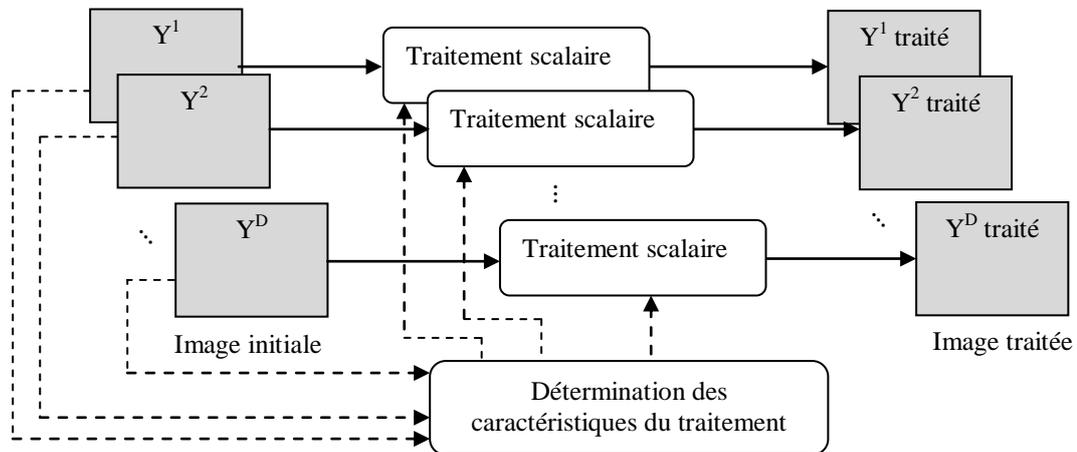


Fig. I.15 : Stratégie de traitement hybride

#### 5 Colorimétrie et images couleur

De nombreuses caractéristiques de la vision humaine sont exploitées en imagerie couleur. Ainsi, la colorimétrie, qui a pour but d'associer à chaque stimulus visuel élémentaire une mesure de stimulus couleur, est fondée sur la théorie trichromatique. Cette théorie s'appuie sur les travaux de Yong et Maxwell au XIX<sup>e</sup> siècle (voir [17-19]) et se base sur un principe expérimental d'égalisation de lumières colorées par mélange ou superposition de trois lumières primaires (rouge, verte et bleue). Les expériences de Guild [20] et Wright [21] ont effectivement démontrées qu'il est possible d'apparier (ou d'égaliser) n'importe quel stimulus monochromatique avec un minimum de trois lumières monochromatiques. On parle alors de synthèse additive dans la mesure où on ajoute trois constituants lumineux de base.

Il existe de nombreux systèmes colorimétriques qui satisfont au principe d'égalisation des couleurs et chacun présente des caractéristiques particulières [18, 19, 22]. Nous développerons dans ce qui suit les systèmes utilisés dans cette thèse.

### 5.1 Espace RGB (RGB : Red, Green and Blue) (ou RVB : Rouge, Vert et Bleu)

En traitement d'images, l'espace RGB reste de loin le plus utilisé et le plus simple à manipuler. Toute couleur  $C$  de coordonnées  $(r, g, b)$  peut être reproduite par addition des trois primaires  $R, G$  et  $B$ .

$$C = r.R + g.G + b.B \quad (I.5)$$

La définition de l'espace de représentation de la couleur dépend des primaires et du blanc de référence. De nombreux systèmes RGB ont été définis par différents organismes et sont utilisés dans plusieurs domaines d'application. Le plus utilisé des espaces RGB est celui introduit par la CIE en 1931 qui est définie en référence à l'illuminant E [17]. Celui-ci est construit à partir de trois primaires monochromatiques de couleurs rouge, verte et bleue. Les longueurs d'ondes associées à chacune de ces primaires sont initialement définies par Guild [20] et sont 700.0 nm pour le rouge, 546.1 nm pour le vert et 435.8 nm pour le bleu [23]. Il traduit le comportement d'un observateur de référence qui est vérifié par 90% des observateurs [23]. Les fonctions colorimétriques (ou composantes trichromatiques spectrales) de cet observateur dit standard sont données par la figure I.16.

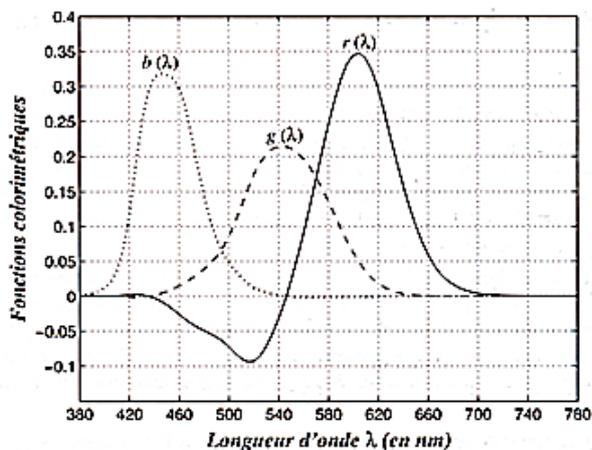


Fig. I.16 : Fonctions colorimétriques de l'observateur du système RGB CIE1931

### 5.2 Espace XYZ

L'espace couleur XYZ a été défini afin de corriger certains défauts de l'espace RGB défini par la CIE 1931. En effet l'existence d'une partie négative dans leurs spectres, ne permet pas de reproduire un certain nombre de couleurs par superposition des trois spectres, ce qui contredit le principe de la synthèse additive. L'espace couleur XYZ résulte des travaux de Judd [24], il est constitué de trois primaires  $X, Y$  et  $Z$ , dites virtuelles tel que :

- les triplets décrivent chaque couleur en fonction de ses primaires qui ont tous des valeurs positives pour les spectres visibles (voir Fig. I.17),

- la fonction  $Y(\lambda)$  représente approximativement la sensibilité de l'œil humain à la luminosité. Par conséquent, la composante  $Y$  est usuellement considérée comme la composante luminance du spectre incident,

- tout spectre d'égale énergie est associé à un triplet dont toutes les composantes sont égales.

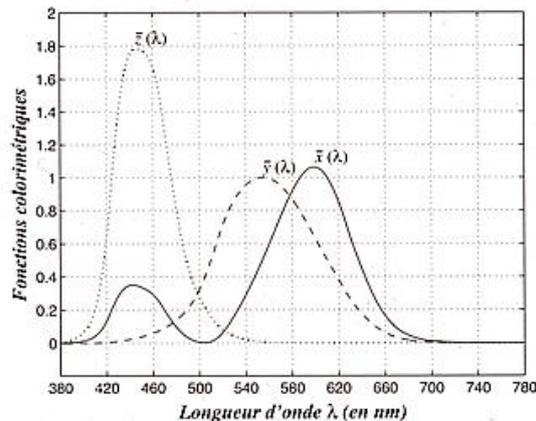


Fig. I.17 : Fonction colorimétriques du système XYZ de l'observateur de référence CIE1931

Le passage de l'espace RGB à l'espace XYZ s'effectue simplement grâce à une transformation linéaire pouvant être interprétée comme un changement de base. Cette transformation prend en compte les coordonnées chromatiques des primaires R, G et B et de l'illuminant de référence choisi. Par conséquent, plusieurs matrices de passage ont vu le jour (voir [17, 24, 25]). Le passage du système de primaires RGB CIE 1931 au système de primaires XYZ CIE 1931 est donné par l'équation :

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 2.7690 & 1.7518 & 1.1300 \\ 1.0000 & 4.5907 & 0.0601 \\ 0.0000 & 0.0565 & 5.5943 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{I.6})$$

Afin de pouvoir représenter et analyser assez simplement, dans un repère géométrique à deux dimensions les coordonnées  $X$ ,  $Y$  et  $Z$ , on utilise en général les coordonnées normalisées  $x$ ,  $y$  et  $z$  suivantes :

$$\begin{cases} x = \frac{X}{X+Y+Z} \\ y = \frac{Y}{X+Y+Z} \\ z = \frac{Z}{X+Y+Z} \end{cases} \quad (\text{I.7})$$

Etant donné que la somme des trois coordonnées vaut 1,  $z$  se déduit directement des deux autres, ceci justifie pourquoi cette coordonnée n'est généralement pas utilisée.

L'espace  $xyY$  est communément utilisé pour représenter l'espace trichromatique. Il est le plus généralement décrit par une représentation 2D à  $Y$  constant. Une représentation à 3D est toujours possible. (Voir Fig. I.18)

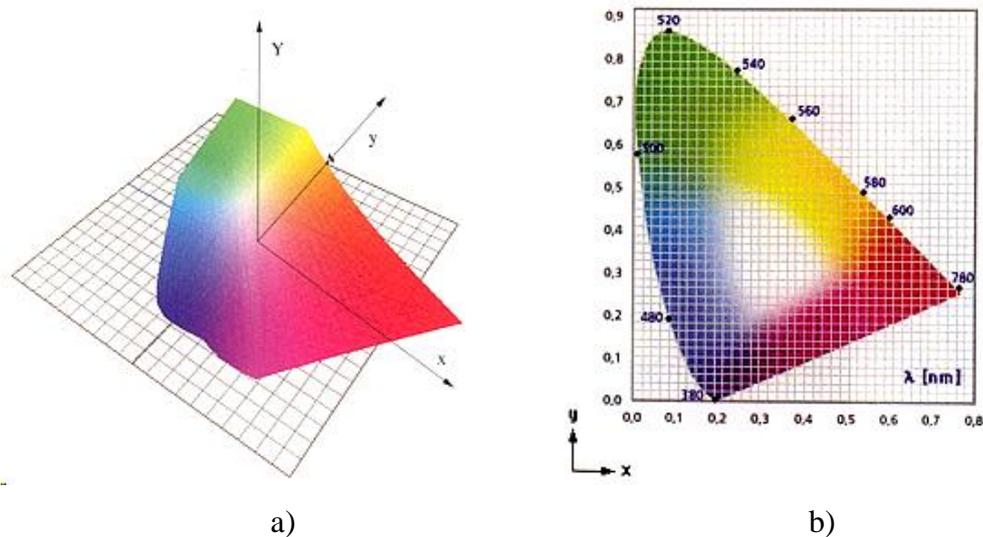


Fig. I.18: Représentation du diagramme de chromaticité  $xyY$   
 a) Espace  $xyY$  en trois dimensions b) Coupe d'un plan  $xy$  à  $Y$  constant de l'espace  $xyY$ . La droite qui lie le rouge (780 nm) au bleu (380 nm) est appelée la droite des pourpres

### 5.3 Système de cône hexagonal HSV

Le système HSV, connu aussi sous le nom du système de cône hexagonal, présente deux variantes : le modèle de cône hexagonal simple et le modèle hexagonal double (Fig. I.19). Ils représentent la couleur sous forme d'un triplet : Teinte  $H$  (Hue), Saturation  $S$  et luminosité  $V$  (Value).

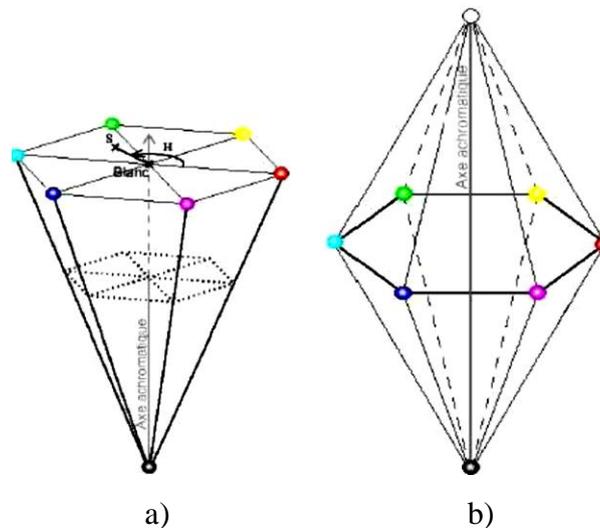


Fig. I.19 : Modèle de cône hexagonal  
 a) modèle de cône hexagonal simple b) modèle de cône hexagonal double dit aussi HSL

Dans le cas du modèle simple les transformations sont effectuées comme suit :

$$\begin{aligned}
 V &= \max (R, G, B) \\
 S &= \frac{V - \min (R, G, B)}{V} \\
 H &= \begin{cases} \frac{G-B}{V - \min (R, G, B)} & \text{si } V = R \\ 2 + \frac{B-R}{V - \min (R, G, B)} & \text{si } V = G \\ 4 + \frac{R-G}{V - \min (R, G, B)} & \text{si } V = B \end{cases} \quad (\text{I.8})
 \end{aligned}$$

Le modèle de cône hexagonal double est construit sur le même principe que le modèle précédent. Sachant que la formule pour calculer la teinte H est la même que l'équation I.8, la valeur V et la saturation S sont calculées comme suit :

$$\begin{aligned}
 S &= \begin{cases} \frac{\max(R, G, B) - \min(R, G, B)}{2V} & \text{si } V \leq \frac{V_{\max}}{2} \\ \frac{\max(R, G, B) - \min(R, G, B)}{2(V_{\max} - V)} & \text{si } V > \frac{V_{\max}}{2} \end{cases} \\
 V &= \frac{\max(R, G, B) + \min(R, G, B)}{2}
 \end{aligned} \quad (\text{I.9})$$

#### 5.4 Espace CIE L\*a\*b\*

Dans l'année 1976, la CIE a introduit un système perpétuellement uniforme, le système CIELAB ou L\*a\*b\*. Le système L\*a\*b\* est une approximation de l'espace d'Adams-Nickerson dans lequel l'amplitude perceptuelle de la couleur est définie en termes d'échelles de couleurs opposées couvrant l'intégralité du spectre visible par l'œil humain. Le passage au modèle L\*a\*b\* s'obtient à partir du modèle XYZ. Les composantes de cet espace sont données par les équations suivantes :

$$\begin{aligned}
 L^* &= \begin{cases} 116 \left( \frac{Y}{Y_0} \right)^{\frac{1}{3}} - 16 & \text{si } \frac{Y}{Y_0} > 0.008856 \\ 903.3 \frac{Y}{Y_0} & \text{si } \frac{Y}{Y_0} \leq 0.008856 \end{cases} \\
 a^* &= 500 \left[ f \left( \frac{x}{x_0} \right) - f \left( \frac{y}{y_0} \right) \right] \\
 b^* &= 300 \left[ f \left( \frac{y}{y_0} \right) - f \left( \frac{z}{z_0} \right) \right]
 \end{aligned} \quad (\text{I.10})$$

avec :

$$f(x) = \begin{cases} x^{\frac{1}{3}} & \text{si } x > 0.008856 \\ 7.787x + \frac{16}{116} & \text{si } x \leq 0.008856 \end{cases} \quad (\text{I.11})$$

Cette transformation se définit en fonction des coordonnées trichromatiques ( $X_0, Y_0, Z_0$ ) de l'illuminant de référence. Le fait de représenter la luminosité par une racine cubique a été démontré par les expériences psycho-visuelles menées sur l'appréciation des distances entre

des stimuli monochromatiques. De plus, le seuil de 0.008856 est choisi afin de supprimer le caractère trop abrupt des variations de  $(Y/Y_0)^{1/3}$  autour de 0. Cette racine cubique est néanmoins très intéressante pour le rapprochement du fonctionnement de l'œil humain. En effet, l'introduction du rapport  $Y/Y_0$  permet de simuler grossièrement l'adaptation de l'œil humain à une luminosité donnée. Dans le cas de faibles luminances (vision de nuit par exemple), la pente de la courbe est assez importante marquant l'intervention des bâtonnets. Au contraire, un effet de saturation peut être observé pour les fortes luminances du fait de la saturation des récepteurs de l'œil.

L'espace  $L^*a^*b^*$  appartient à la famille des systèmes antagonistes puisque  $L^*$  met en évidence l'opposition des couleurs noir-blanc (0 pour le noir et 100 pour le blanc),  $a^*$  mesure l'opposition des couleurs vert-rouge par une valeur comprise entre  $-100$  et  $+100$  ( $a^*$  est positif si la couleur contient du rouge, négatif si la couleur contient du vert et nulle si aucun des deux), et  $b^*$  mesure l'opposition des couleurs bleu-jaune par une valeur comprise entre  $-100$  et  $+100$  ( $b^*$  est positif si la couleur contient du jaune, négatif si la couleur contient du bleu et nulle si aucun des deux) (voir Fig. I.20).

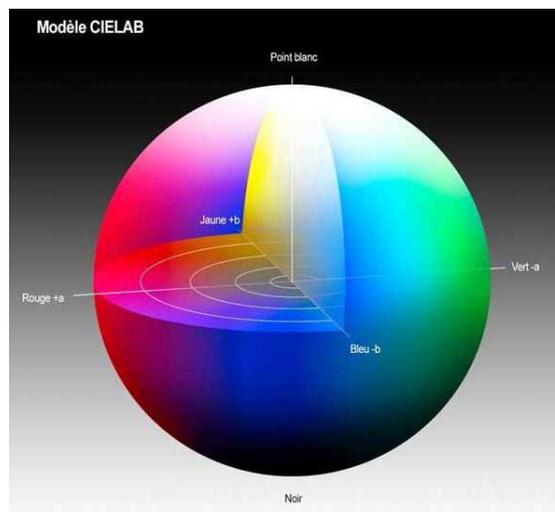


Fig. I.20 : Représentation spatiale de l'espace colorimétrique CIE  $L^*a^*b^*$

## 6 Télédétection et images MSG2

### 6.1 Télédétection spatiale et satellite météorologique

La télédétection est la technique qui, par l'acquisition d'images, permet d'obtenir de l'information sur la surface de la terre sans contact direct avec celle-ci. La télédétection englobe tout le processus qui consiste à capter, à enregistrer l'énergie d'un rayonnement électromagnétique émis ou réfléchi, à traiter et à analyser l'information pour ensuite mettre en

application cette information. En télédétection, l'énergie diffusée ou émise par la cible est recueillie à distance par des dispositifs appelés capteurs.

Les satellites météorologiques sont souvent géostationnaires c'est-à-dire maintenus sur des orbites au cours de laquelle ils sont toujours dans la même position par rapport à la terre en rotation. Parmi eux il y'a principalement le satellite Meteosat maintenu en orbite par l'Europe (Eumetsat), GOES par les États Unis, GMS par le Japon, FY-2B par la Chine, GOMS par la Russie et INSAT par l'Inde. Ils sont placés au dessus de l'équateur à une altitude de 35800km. Leur vitesse de rotation est la même que celle à laquelle la terre tourne autour de son axe de rotation. La figure I.21 donne la position de chacun de ces satellites.

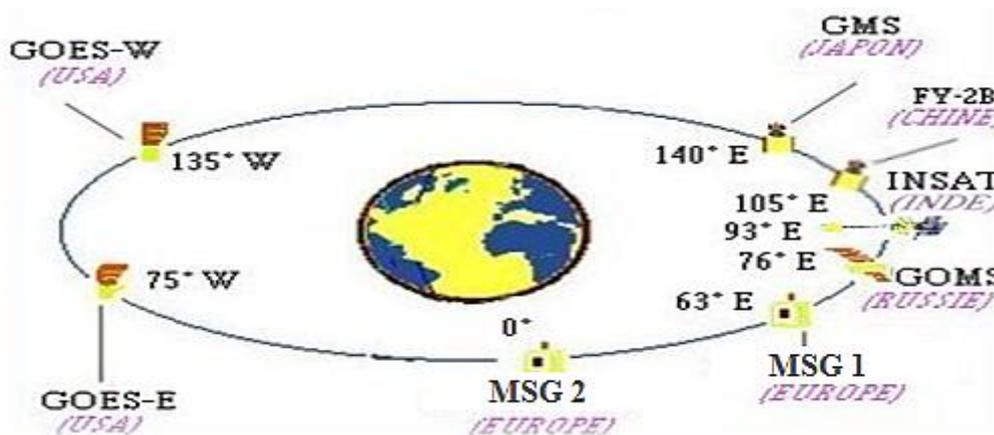


Fig. I.21 : Emplacement des satellites géostationnaires autour de la terre

## 6.2 Satellites Meteosat

Le satellite géostationnaire européen Meteosat, est placé en orbite équatoriale à 35800 km d'altitude environ. Sa position nominale se situe à l'intersection du méridien de Greenwich et de l'Equateur, ce qui en fait un instrument particulièrement adapté à l'étude du continent africain et européen. Il s'agit d'un programme commun de l'Agence spatiale européenne (ESA) et de l'Eumetsat, organisation européenne de satellites météorologiques [6].

Le premier satellite, Meteosat 1, fut lancé en Novembre 1977 pour une mission expérimentale, il a été suivi de Meteosat 2 en 1981. Ces deux premiers satellites Meteosat couvrent la période dite pré-opérationnelle. La période opérationnelle débutera avec Meteosat 3. Cinq autres satellites furent lancés en 1988, 1989, 1991, 1993 et 1997. L'ensemble de ces sept satellites constituent la première génération de Meteosat.

La seconde génération des satellites Meteosat débute avec le lancement du satellite MSG1 (Meteosat Seconde Génération 1) le 28 août 2002. Celui-ci est rejoint en orbite par un autre satellite MSG2 lancé le 21 décembre 2005. MSG1 est maintenu en orbite et servira alors

de satellite de secours en cas de panne de MSG2. La durée de vie de ces satellites est de sept ans. Cette génération de satellites dispose de radiomètres imageurs sophistiqués contenant douze canaux spectraux au lieu de trois pour la génération précédente, et fournissant des images tous les quarts d'heure au lieu d'une demi-heure.

### 6.3 Images MSG2

Les trois canaux des satellites Meteosat de première génération sont remplacés par douze canaux dans les satellites MSG (voir Fig. I.22). Deux canaux sont dans le domaine du visible, deux dans le proche et moyen infrarouge, deux dans le vapeur d'eau, cinq dans l'infrarouge et le dernier un peu particulier est le visible haute résolution nommé HRV. La résolution des canaux infrarouge passe de 5 Km à 3 Km, tandis que le nouveau canal visible HRV fournit des images de 1 Km de résolution au lieu de 2.5 Km. Le tableau I.1 illustre les différents canaux utilisés par les satellites MSG.

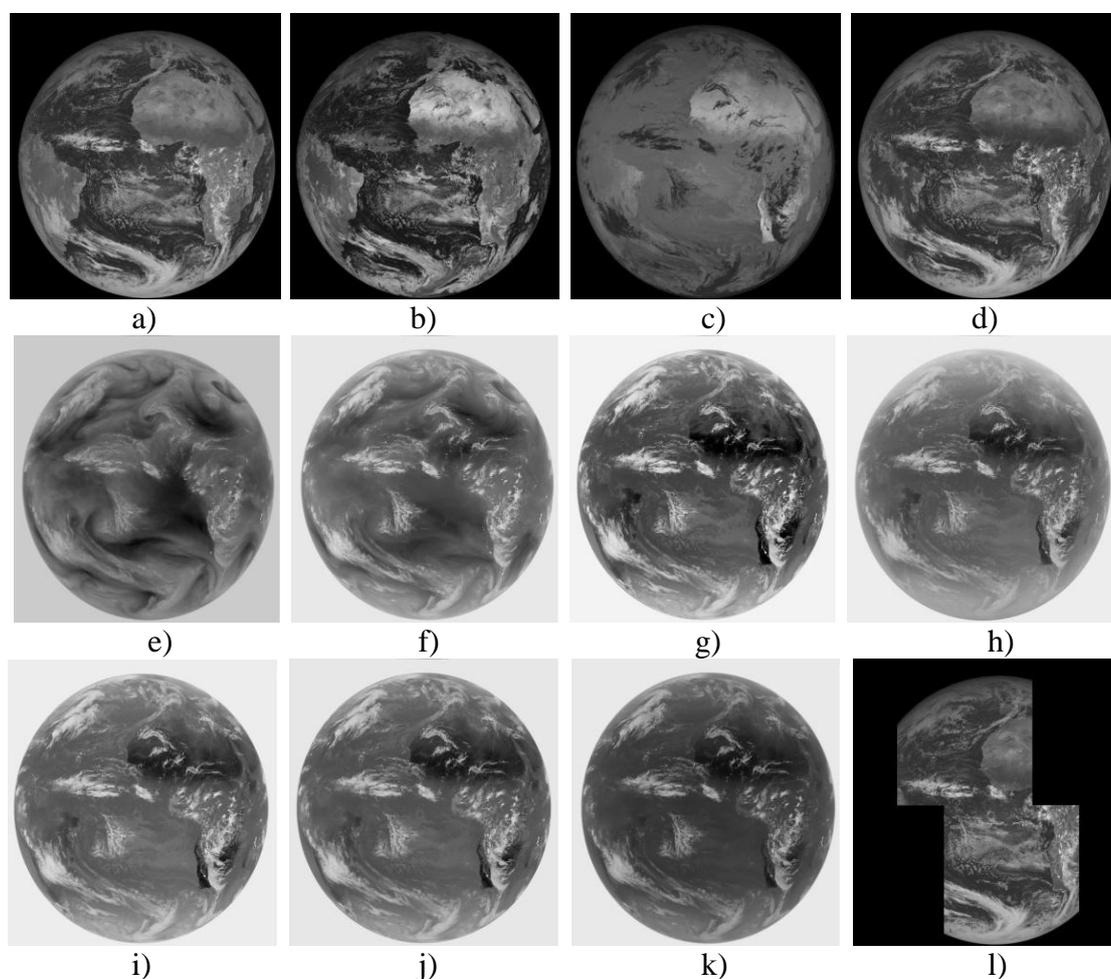


Fig. I.22 : Exemple d'image MSG (12 canaux spectraux) prise le 28 novembre 2002 à 12h00

Les canaux a) VIS0.6 b) VIS0.8 c) NIR1.6 d) NIR3.9 e) WV6.2 f) WV7.3

g) IR8.7 h) IR9.7 i) IR10.8 j) IR12.0 k) IR13.4 l) HRV

Tableau I.1 : Les douze canaux des satellites MSG

| Canal  | bande spectrale           | Résolution spatiale | champ d'observation | Nombre de détecteurs |
|--------|---------------------------|---------------------|---------------------|----------------------|
| HRV    | 0.5 à 0.9 $\mu\text{m}$   | 1 km                | 2500 km             | 9                    |
| VIS0.6 | 0.56 à 0.71 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| VIS0.8 | 0.74 à 0.88 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| NIR1.6 | 1.5 à 1.78 $\mu\text{m}$  | 3 km                | 12500 km            | 3                    |
| NIR3.9 | 3.48 à 4.36 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| WV6.2  | 5.35 à 7.15 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| WV7.3  | 6.85 à 7.85 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| IR8.7  | 8.3 à 9.1 $\mu\text{m}$   | 3 km                | 12500 km            | 3                    |
| IR9.7  | 9.38 à 9.94 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |
| IR10.8 | 9.8 à 11.8 $\mu\text{m}$  | 3 km                | 12500 km            | 3                    |
| IR12.0 | 11 à 13 $\mu\text{m}$     | 3 km                | 12500 km            | 3                    |
| IR13.4 | 12.4 à 14.4 $\mu\text{m}$ | 3 km                | 12500 km            | 3                    |

Le domaine visible est constitué des canaux VIS0.6 et VIS0.8 telle que la bande passante de chacun d'eux est de 0.56 $\mu\text{m}$  à 0.71 $\mu\text{m}$  pour le VIS0.6 et 0.74 $\mu\text{m}$  à 0.88 $\mu\text{m}$  pour le VIS0.8. Ces deux canaux sont utilisés pour la détection et le suivi des nuages durant la journée. La réflectivité de la végétation "verte" est plus importante pour le VIS0.8. La combinaison des deux canaux est un outil important pour déterminer l'indice de végétation.

Le domaine proche infrarouge est constitué du canal NIR1.6. Il a une bande passante qui est comprise entre 1.5 $\mu\text{m}$  et 1.78 $\mu\text{m}$ . Il permet de distinguer la neige au sol des nuages. Il permet également de différencier les nuages de glaces des nuages d'eau liquide. Il donne des informations sur la présence des aérosols.

Le domaine moyen infrarouge est constitué du canal NIR3.9. Il a une bande passante qui est comprise entre 3.48 $\mu\text{m}$  et 4.36 $\mu\text{m}$ . Il permet de distinguer les nuages bas et le brouillard de nuit. Il est fortement utilisé pour la détection des feux de forêt.

Le domaine d'absorption vapeur d'eau est constitué des canaux WV6.2 et WV7.3. La bande passante de WV6.2 est de 5.35 $\mu\text{m}$  à 7.15 $\mu\text{m}$ , pour le WV7.3 elle est de 6.85 $\mu\text{m}$  à 7.85 $\mu\text{m}$ . Le premier nous renseigne sur la haute troposphère et le second sur la moyenne troposphère. Ils fournissent des traceurs pour la détermination des vents atmosphériques. Ils permettent de connaître l'altitude des nuages et le suivi de ces derniers, et donnent aussi des informations sur l'humidité de toutes les régions terrestres.

Le domaine de l'infrarouge est constitué des canaux IR8.7, IR9.7, IR10.8, IR12.0 et IR13.4. Pour l'IR8.7, sa bande passante est de 8.3 $\mu\text{m}$  à 9.1 $\mu\text{m}$ . Il renseigne sur les cirrus fin et il permet de différencier les nuages d'eau liquide des nuages de glace. L'IR9.7 a sa bande passante qui est comprise entre 9.38 $\mu\text{m}$  et 9.94 $\mu\text{m}$ . Il donne une information pour le calcul de

l'ozone total et pour sa variation diurne. Il fournit également des traceurs pour la détermination des vents atmosphériques. L'IR10.8 a sa bande passante qui est de  $9.8\mu\text{m}$  à  $11.8\mu\text{m}$ . Il renseigne sur la température de surface. L'IR12.0 a sa bande passante comprise entre  $11\mu\text{m}$  à  $13\mu\text{m}$ . Il renseigne sur les nuages et l'humidité dans les basses couches de l'atmosphère. L'IR13.4 a sa bande passante qui est comprise entre  $12.4\mu\text{m}$  et  $14.4\mu\text{m}$ , il permet de mesurer la température des basses couches.

Le dernier canal HRV est un canal du domaine de visible, mais il est de très haute résolution. Il permet de détecter les vecteurs vents dans la haute résolution dans le visible. Combiné au canal VIS 0.6 du visible, il permet la détection et le suivi des nuages.

Tous les canaux ont un champ d'observation de 12500 Km. Chacun des 11 canaux (à part le HRV) est équipé de trois détecteurs, chaque détecteur peut balayer 1250 lignes, ce qui donne 3750 lignes d'images par canal. Le canal HRV balaye la moitié du disque Est-Ouest tout en réalisant un échantillonnage de 1Km à l'aide de 9 détecteurs.

## 7 Conclusion

Dans ce chapitre, nous avons défini les différents types d'images multi-composantes c'est-à-dire des images caractérisées par le fait que la valeur d'un pixel n'est plus représentable par un scalaire, mais par plusieurs composantes. Ces composantes peuvent être homogènes comme dans le cas des images multi-temporelles ou polarimétriques, quasi-homogènes comme dans le cas des images multi-spectrales ou couleur RGB, ou hétérogènes comme dans le cas des images IRM ou couleur HSV. Le traitement et l'analyse envisagés sur ces images peuvent influencer la nature de leur représentation qui peut être multi-scalaire, volumique ou vectoriel. La mise en œuvre d'un traitement sur les images multi-composantes peut se faire manière scalaire, marginal, vectoriel ou hybride.

La première approche consiste à fusionner, sous forme mono-composante, les différentes composantes de l'image multi-composante, puis à traiter l'image en niveaux de gris ainsi obtenue. L'approche qualifiée de marginale, consiste à appliquer un même traitement sur chacune des composantes de l'image de manière indépendante. L'approche vectorielle consiste au contraire à considérer chaque pixel comme un vecteur et à n'appliquer qu'un seul traitement sur l'image. L'approche hybride commute entre les stratégies marginales et vectorielles afin de tirer le meilleur parti de chaque approche.

Puisque dans les chapitres qui suivent différents traitements sont appliquées à des images couleur et Meteosat, ces deux types d'images multi-composantes ont été alors largement développés dans ce chapitre.

## Chapitre II

# Segmentation marginale et vectorielle des images multi-composantes

### 1 Introduction

La segmentation est une étape importante dans un système de traitement d'images. Elle n'est pas une finalité en soi, mais une étape située en amont de dispositifs de reconnaissance des formes, d'interprétation d'images, de reconstruction tridimensionnelle, de codage, de compression, etc. Elle consiste à extraire de l'image des primitives, soit de type contour, soit de type région. A la différence du prétraitement (filtrage), qui transforme l'image multi-composante en une autre image multi-composante, la segmentation transforme l'image multi-composante soit en une image binaire contours/non contours, soit en une image de régions pour laquelle tous les pixels d'une même région sont affectés d'une même étiquette.

Comme vu dans le chapitre précédent, il existe trois approches possibles pour le traitement des images multi-composantes en vue de la segmentation : approche scalaire, approche marginale et approche vectorielle. Notre contribution s'inscrit dans le cadre de la segmentation marginale des images multi-composantes. Dans cette stratégie de segmentation, chaque composante de l'image est traitée séparément donnant ainsi autant de cartes des étiquettes que l'image contient de composantes. Les cartes des étiquettes doivent être fusionnées pour fournir le résultat final. Nous proposons, dans ce chapitre, une nouvelle technique de fusion de ces cartes. Les résultats obtenus seront comparés à ceux de la segmentation vectorielle en utilisant plusieurs algorithmes de classification tels que les algorithmes EM, K-means, FCM et SOM.

### 2 Stratégie de segmentation vectorielle

Dans l'approche vectorielle chaque pixel est traité comme un vecteur dans l'espace de représentation de l'image multi-composante. Les primitives sont extraites en tenant compte de toutes les composantes de l'image à la fois. Le passage de l'image multi-composante à

l'image mono-composante donnant l'image des étiquettes se fait directement par la technique de segmentation utilisée. Aucune fusion n'intervient dans cette stratégie, ni avant traitement sur les différentes composantes de l'image, ni après traitement sur les cartes des étiquettes. En pratique, la majeure partie des techniques utilisées pour la segmentation des images en niveaux de gris peut être utilisée en généralisant celles-ci à plusieurs dimensions. La littérature foisonne de méthodes de segmentation, mais aucune ne permet d'obtenir des résultats satisfaisants pour toutes les images. Le choix d'une méthode dépend souvent du type de l'image, de la présence ou pas de texture et de bruit, mais surtout du but recherché. Il existe de nombreuses méthodes de segmentation, que l'on peut regrouper en quatre principales classes :

- segmentation fondée sur les régions (en anglais : *region-based segmentation*). On y trouve par exemple la croissance de région (en anglais : *region-growing*) et la décomposition/fusion (en anglais : *split and merge*),
- segmentation fondée sur les contours (en anglais : *edge-based segmentation*),
- segmentation fondée sur la classification ou le seuillage des pixels en fonction de leur intensité (en anglais : *classification, thresholding*),
- segmentation fondée sur la coopération entre les trois premières méthodes.

Nous nous concentrerons dans ce travail uniquement sur les méthodes de segmentation fondées sur la classification car notre contribution intervient dans ce type de méthodes. Formellement, la segmentation fondée sur la classification consiste à partitionner les pixels d'une image en sous-ensembles disjoints non vides contenant des pixels homogènes au sens d'un prédicat. Les pixels sont alors assignés aux différentes classes construites, une analyse en composantes connexe permet de différencier les différentes régions qui composent l'image. En effet, une région est définie comme un ensemble de pixels connexes assignés à une même classe, donc dans une même classe peuvent subsister plusieurs régions. De nombreuses méthodes de segmentation fondées sur la classification existent. Parmi celles-ci, nous énumérons un certain nombre d'entre elles et nous les appliquerons à des images multi-composantes et plus particulièrement à des images couleur et à des images Meteosat en utilisant des approches vectorielle et marginale.

### **3 Stratégie de segmentation marginale et proposition d'un algorithme de fusion**

#### **3.1 Etat de l'art de la segmentation marginale**

Dans l'approche marginale, les différentes composantes de l'image sont traitées séparément. Les différentes cartes des étiquettes, obtenues pour chaque composante, sont fusionnées ensuite. La segmentation marginale se fait alors en deux étapes qui sont :

- segmentation monochrome indépendante des différentes composantes de l'image et
- fusion des cartes des étiquettes obtenues dans l'étape précédente.

En ce qui concerne la première étape, toutes les techniques de segmentation des images en niveaux de gris peuvent être utilisées. Celles utilisées pour les images multi-composantes vectorielles peuvent aussi être employées en réduisant la dimension des vecteurs représentant les pixels à un. La deuxième étape de cette stratégie est très délicate. On trouve peu de travaux qui traitent de la fusion des images des étiquettes. Ces travaux traitent généralement de la fusion d'images contour/non-contour obtenues en utilisant une segmentation type contour sur les différentes composantes de l'image. La fusion des cartes contour/non-contour est souvent réalisée par des opérateurs logiques tel le "ou" et le "et" logique [17]. Dans le cas où les cartes obtenues en segmentant les composantes de l'image de manière marginale sont des cartes étiquettes, le problème de la fusion devient plus accru. Plusieurs auteurs [37, 38, 39] réalisent la fusion en utilisant la théorie de l'évidence qui, initialement introduite par Dempster [40, 41], fut reprise dans un formalisme plus mathématique par Shafer [42]. Elle est également connue dans la littérature sous le nom de "théorie des croyances", "théorie des plausibilités" ou bien encore "théorie de Dempster-Shafer". L'une des difficultés majeures de cette théorie réside dans le choix de la modélisation des fonctions de masse. La fonction de masse étant le degré de confiance accordée à un élément focale qui est une réunion des éléments (un ou plusieurs) originaux qu'on souhaite fusionner. Dans [43], on peut trouver une comparaison de trois modèles d'évidence utilisés dans le cadre de la segmentation d'images. Une autre difficulté de cette technique réside dans l'établissement des liens de compatibilité entre les hypothèses. En effet, les différentes composantes de l'image étant segmentées et étiquetées indépendamment, les cartes des étiquettes sont alors considérées comme des sources indépendantes. Il n'y a aucune garantie qu'une même étiquette ne soit attribuée à la même zone d'intérêt dans les différentes cartes. Afin de réaliser la fusion par la théorie de l'évidence, une correspondance des étiquettes doit être réalisée au préalable.

D'autres auteurs ont traité le problème de la fusion des cartes des étiquettes de manière différente. Par exemple, Kurugollu et al. [44] commencent l'étape de fusion en concordant les étiquettes dans les différentes cartes. Ils construisent ensuite d'autres cartes des étiquettes, qu'ils nomment cartes de chevauchement en attribuant la même valeur aux étiquettes présentant un maximum de chevauchement dans les cartes précédentes, prises deux à deux. Au final, ils étiquettent la carte fusion en faisant chevaucher toutes les cartes de chevauchement à la fois. Lézoray et Charrier [45], quant à eux, commencent l'étape de fusion en éliminant dans chaque carte les régions dont la dimension est inférieure à un élément

structurant de taille  $3 \times 3$ . Par la suite une carte fusion est générée par superposition des précédentes, le nombre d'étiquettes dans cette dernière se trouve alors augmenté. Au final, la fusion de régions est réalisée en utilisant un graphe d'adjacence spatiale des différentes régions de la carte. Les régions (les nœuds) adjacentes dont la similarité est importante sont alors fusionnées.

### 3.2 Méthode de fusion proposée

En ce qui nous concerne, nous proposons une nouvelle technique de fusion des images des étiquettes. Elle se déroule en quatre étapes et ces différentes étapes sont illustrées en figure II.1. Pour distinguer les différents changements au cours des étapes, une illustration sur une image couleur est produite en figure II.4.

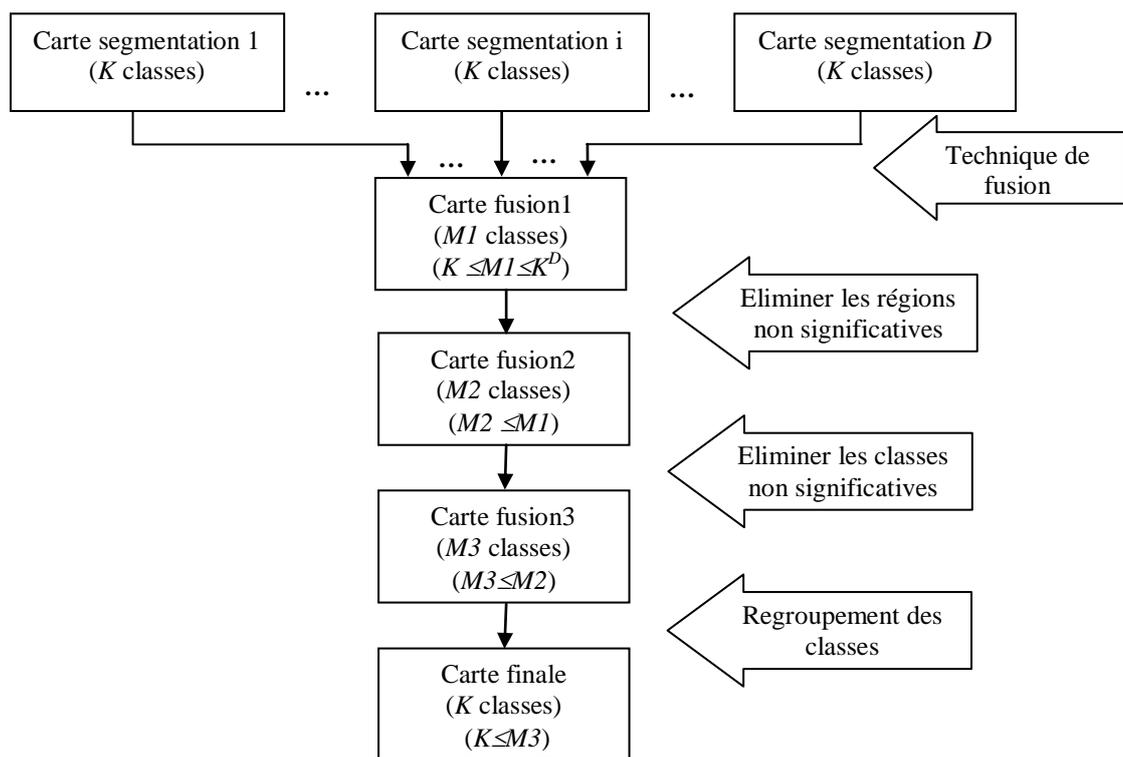


Fig. II.1 : Schéma des étapes de la fusion et de regroupement

La première étape consiste à créer, à partir des cartes des étiquettes obtenues lors de l'étape de segmentation marginale qu'on appellera par la suite "cartes segmentation", une nouvelle carte en réalisant leur fusion. Pour générer cette nouvelle carte, appelée "carte fusion1", nous affectons à la même combinaison de classes dans les "cartes segmentation" le même numéro dans la "carte fusion1" et pour des combinaisons différentes, nous affectons des numéros différents. Ainsi, si nous avons initialement  $K$  classes dans chaque "carte segmentation", au final dans la carte "fusion1", nous obtenons  $M1$  classes tel que  $K \leq M1 \leq K^D$

avec  $D$  est le nombre de composantes de l'image. Ce nombre  $M1$  est généralement important, il convient alors de le réduire. Pour ce faire, les régions et les classes non significatives seront éliminés avant d'opérer des regroupements de classes proprement dit jusqu'à obtention du nombre de classes souhaité. Les petites régions dite aussi trous, sont dans notre cas des régions dont la taille ne dépasse pas 0.06% de la taille de l'image. Dans le cas d'une image de 128x128 pixels, il s'agit de régions dont la dimension est inférieure à 9 pixels. Dans le cas où la taille de l'image est de 256x256 pixels, les régions non significatives seront de taille inférieure à 39 et pour l'image de 512x512 pixels, elles seront inférieures à 157. Ces régions seront jointes à des régions qui leurs sont voisines et qui leurs sont radiométriquement proches. Cette procédure est réalisée en plusieurs étapes qui sont décrites par l'algorithme de la figure II.2.

1. Décomposer la carte fusion1 en carte de régions connexes
2. Rechercher les régions non significatives c.à.d. celles dont la taille est inférieure à 0.06% de la taille de l'image
3. Pour chacune de ces régions :
  - Rechercher les classes qui l'entourent dans le domaine spatial,
  - Fusionner chacune de ces régions à la classe de son entourage qui lui est radiométriquement la plus proche.

Fig. II.2 : Algorithme d'élimination des trous

Ainsi à la fin de cette étape, nous obtenons une autre carte des classes que nous nommons "carte fusion2" qui contient un nombre de classes  $M2$  tel que  $M2 \leq M1$ .

Nous passons par la suite à l'illimitation des petites classes non significatives dont le nombre de pixels ne dépasse pas 0.5 % de la taille de l'image, soit 81 pixels pour une image de taille 128x128, 327 pixels pour une image de taille 256x256 et 1310 pixels pour une image de taille 512x512. Nous allons les fusionner avec des classes dont la taille est plus importante sauf que les composantes connexes de ces petites classes peuvent être jointes à des classes différentes. Cette procédure est aussi réalisée en plusieurs étapes qui sont résumées dans l'algorithme de la figure II.3.

1. Rechercher les classes non significatives c.à.d. celles présentant une taille inférieure à 0.5% de la taille de l'image,
2. Pour chacune de ces classes :
  - La décomposer en région connexes
  - Fusionner chacune de ces régions à la classe de son entourage qui lui est radiométriquement la plus proche.

Fig. II.3 : Algorithme d'élimination des classes non significatives

A la fin de cette procédure, nous obtenons une carte des classes nommée "carte fusion3" dont le nombre de classes  $M3$  est tel que  $M3 \leq M2$ .

Après avoir éliminé les régions et les classes non significatives, la dernière étape de cette technique de fusion consiste à fusionner les classes radiométriquement proches de manière hiérarchique ascendante jusqu'à obtention du nombre de classes  $K$  souhaité (voir Fig. II.4). C'est ce résultat final qui pourra être comparé au résultat obtenu en réalisant une classification vectorielle. Dans les paragraphes qui suivent, nous procédons à la segmentation marginale et vectorielle des images multi-composantes en utilisant quelques techniques de classifications connues.



Fig. II.4 : Illustration des différentes étapes de la fusion sur l'image Fleurs

a) image original b) segmentation marginal de la composante rouge en 4 classes c) segmentation marginal de la composante verte en 4 classes d) segmentation marginal de la composante bleu en 4 classes e) carte fusion1 donnant 44 classes et 3611 régions f) carte fusion2 donnant 32 classes et 801 régions g) carte fusion3 donnant 30 classes et 740 régions h) carte final donnant 4 classes et 32 régions

## 4 Segmentation des images multi-composantes en utilisant l'algorithme EM

### 4.1 Principe de l'algorithme "Expectation-Maximisation"

L'Estimation-maximisation, en anglais "*Expectation-Maximisation*" (EM), est une méthode d'optimisation itérative utilisée pour l'estimation des paramètres par maximisation de la vraisemblance dans le cas de données incomplètes. Nous entendons par données incomplètes le cas où l'on ne dispose d'aucun échantillon d'apprentissage, ou encore d'aucune réalisation de  $X$ . Auquel cas, l'estimation se fait à partir de la seule observation  $Y$ . En segmentation d'image, les données manquantes sont la connaissance des classes.

Son idée de base est qu'au lieu d'utiliser les seules observations  $Y$  et faire des maximisations complexes ou des simulations, on augmente les observations avec des variables cachées  $X$  pour simplifier les calculs et réaliser une série de maximisations simples [46, 47]. Les observations  $Y$  sont donc considérées comme des données incomplètes, auxquelles on rajoute les données manquantes  $X$  pour aboutir aux données complètes  $(X, Y)$ . En effet, le fait que les variables cachées  $X$  contiennent des informations pertinentes pour l'estimation des paramètres  $\theta$ , et  $\theta$  en retour, permettent de retrouver vraisemblablement les valeurs de  $X$ , suggère la stratégie suivante pour l'estimation des paramètres  $\theta$  à partir des seules observations  $Y$  : retrouver les variables cachées à partir d'une estimée initiale de  $\theta$ , ré-estimer  $\theta$  en se basant sur les observations  $Y$  et les variables cachées évaluées  $X$  et réitérer jusqu'à convergence des estimés [48].

Chaque itération de l'algorithme consiste en une étape de calcul de l'espérance suivi d'une étape de maximisation. L'idée de l'algorithme EM est d'utiliser, non pas la densité de probabilité des données incomplètes comme cela est utilisé lors d'une estimation au sens du maximum de vraisemblance, mais la densité de probabilité des données complètes.

La distribution des données complètes est donnée par :

$$p((X, Y) ; \theta) = p(Y ; \theta) p(X/Y ; \theta) \quad (\text{II.1})$$

On souhaite déterminer les paramètres  $\theta$  qui maximisent la Log-vraisemblance des observations  $Y=(y_1, y_2, \dots, y_n)$ . Cette Log-vraisemblance est donnée par :

$$L(Y ; \theta) = \sum_{i=1}^n \log P(y_i ; \theta) \quad (\text{II.2})$$

L'algorithme EM repose sur l'idée qu'il est plus facile de maximiser la Log-vraisemblance complétée donnée par :

$$\begin{aligned}
L((X, Y); \theta) &= \sum_{i=1}^n (\log P(x_i|y_i; \theta) + \log P(y_i; \theta)) \\
&= L(Y; \theta) + \sum_{i=1}^n (\log P(x_i|y_i; \theta))
\end{aligned} \tag{II.3}$$

L'algorithme EM est une procédure itérative basée sur l'espérance des données complétées conditionnellement au paramètre courant  $\theta^{(t)}$ .

Ainsi on aura :

$$E[L((X, Y); \theta)|\theta^{(t)}] = E[L(Y; \theta)|\theta^{(t)}] + E\left[\sum_{i=1}^n (\log P(x_i|y_i; \theta))|\theta^{(t)}\right] \tag{II.4}$$

On montre que la suite définie par  $\theta^{(t+1)} = \arg \max_{\theta} (E[L((X, Y); \theta)|\theta^{(t)}])$  fait tendre  $L(Y, \theta)$  vers un maximum local [48]. On peut donc définir l'algorithme EM de la manière suivante:

1. Initialisation au hasard de  $\theta^{(0)}$
2.  $t=0$
3. Tant que l'algorithme n'a pas convergé, faire
  - étape E : Evaluation de l'espérance  $Q(\theta; \theta^{(t)}) = E[L((X, Y); \theta)|\theta^{(t)}]$
  - étape M : Maximisation  $\theta^{(t+1)} = \arg \max_{\theta} (Q(\theta; \theta^{(t)}))$
  - $t=t+1$
4. Fin

Fig. II.5 : L'algorithme EM

#### 4.2 L'algorithme EM dans le cas des mélanges gaussiens

Dans le cas des mélanges gaussiens, un échantillon  $y_i \in \mathbb{R}^D$  avec ( $y_i \in Y$ ) est en réalité issu de  $K$  groupes (classes)  $C_k$  (avec  $k=1, 2, \dots, K$ ). Chaque classe  $C_k$  suit une loi  $P$  de paramètres  $\theta_k$  et dont les proportions sont données par un vecteur  $(\pi_1, \pi_2, \dots, \pi_K)$ . Ces proportions sont également inconnues et doivent être estimées sous les contraintes :

$$\begin{cases} \pi_k \in ]0,1] \\ \sum_{k=1}^K \pi_k = 1 \end{cases} \tag{II.5}$$

Soit  $\Phi = (\pi_1, \dots, \pi_K, \theta_1, \dots, \theta_K)$  le vecteur des paramètres à estimer avec  $\theta_k$  qui représente la moyenne  $\bar{y}_k$  et la matrice de covariance  $\Sigma_k$  du  $k^{ième}$  groupe gaussien. La fonction de densité de probabilité  $P(y_i)$  en un point  $y_i$  est donnée par la relation :

$$P(y_i) = \sum_{k=1}^K \pi_k P(y_i; \theta) \tag{II.6}$$

La log-vraisemblance des données relativement à  $\Phi$  est alors donnée par :

$$L(Y, \Phi) = \sum_{i=1}^n \log \left( \sum_{k=1}^K \pi_k P(y_i; \theta) \right) \quad (\text{II.7})$$

$$\text{En posant } z_{ik} = \begin{cases} 1 & \text{si } y_i \in C_k \\ 0 & \text{sinon} \end{cases}, \quad (\text{II.8})$$

La log-vraisemblance des données complétées s'écrit :

$$L((X, Y), \Phi) = \sum_{i=1}^n \sum_{k=1}^K z_{ik} \log(\pi_k P(y_i; \theta)) \quad (\text{II.9})$$

Les paramètres qui maximisent l'espérance de cette Log-vraisemblance vérifient alors les équations :

$$\pi_k = \frac{1}{n} \sum_{i=1}^n P(C_k | y_i) \quad (\text{II.10})$$

$$\bar{y}_k = \frac{\sum_{i=1}^n P(C_k | y_i) y_i}{\sum_{i=1}^n P(C_k | y_i)} \quad (\text{II.11})$$

$$\Sigma_k = \frac{\sum_{i=1}^n P(C_k | y_i) (y_i - \bar{y}_k)(y_i - \bar{y}_k)^T}{\sum_{i=1}^n P(C_k | y_i)} \quad (\text{II.12})$$

Où  $P(C_k | y_i)$  représente l'estimation de la probabilité a posteriori d'être en présence d'une observation  $y_i$  de la classe  $C_k$ . Elle est obtenue par la formule de Bayes :

$$P(C_k | y_i) = \frac{P(y_i | C_k) \pi_k}{\sum_{k=1}^K P(y_i | C_k) \pi_k} \quad (\text{II.13})$$

Les étapes de l'algorithme EM dans le cas d'un mélange gaussien est alors résumé dans la figure ci-dessous [49].

1. Initialisation des paramètres  $\Phi^{(0)}(\pi_k^{(0)}, \bar{y}_k^{(0)} \text{ et } \Sigma_k^{(0)})$  avec  $k=1, \dots, K, t=0$
2. Tant que l'algorithme n'a pas convergé, faire
  - étape E : pour  $k=1, \dots, K$  et pour  $i=1, \dots, n$  ; calculer les probabilités a posteriori  $P(C_k | y_i)$ , par l'équation II.13.
  - étape M : pour  $k=1, \dots, K$  ; calculer les nouvelles valeurs  $\pi_k^{(t+1)}, \bar{y}_k^{(t+1)}$  et  $\Sigma_k^{(t+1)}$  à l'aide des équations II.10, II.11 et II.12.
  - $t=t+1$
3. Fin

Fig. II.6 : L'algorithme EM dans le cas d'un mélange gaussien

L'algorithme EM découle des équations de vraisemblance et possède de bonnes propriétés statistiques. Wu [50] a établi que, sous des conditions suffisantes de régularité, l'algorithme

EM assure une convergence vers un maximum local de la vraisemblance. Cependant, l'algorithme EM possède un certain nombre de limitations. La première limitation est que la valeur de l'estimateur à la convergence peut être fortement dépendante de l'initialisation. Une autre limitation de l'algorithme EM est que la convergence peut être lente et l'algorithme peut même se trouver bloqué dans un point selle de la vraisemblance.

### **4.3 Stratégies d'initialisation de l'algorithme EM et proposition d'une méthode d'initialisation**

Pour pallier la limitation de la dépendance de la solution vis à vis de l'initialisation, il est courant dans la pratique de lancer plusieurs EM pour quelques itérations depuis des initialisations aléatoires et de choisir les valeurs de  $\theta$  associées à la plus grande vraisemblance comme initialisation d'un autre EM qui lui itèrera jusqu'à convergence. McLachlan et Peel [51] proposent une autre stratégie d'initialisation de  $\theta$  dans le cas de mélanges gaussien. Ils proposent de fixer les proportions du mélange à des valeurs égales et de générer les moyennes du mélange suivant une loi normale de moyenne et de matrice de covariance qui sont respectivement la moyenne et la matrice de covariance de l'échantillon entier. Les matrices de covariance des composantes du mélange seront également initialisées à la matrice de covariance de l'échantillon entier. En pratique, la stratégie consistant à initialiser à plusieurs reprises de façon aléatoire et à choisir ensuite l'initialisation associée à la plus grande vraisemblance de  $\theta$  est le plus souvent utilisée.

Pour ce qui nous concerne, nous proposons une stratégie d'initialisation différente. Il s'agira de sélectionner manuellement, pour chaque classe, un certain nombre de pixels afin de déterminer la moyenne estimée de cette classe ainsi que sa matrice de covariance. Les proportions du mélange seront égales.

La figure II.7 présente les résultats de la segmentation vectorielle en utilisant une initialisation aléatoire et l'initialisation proposée. Nous remarquons que le résultat obtenu avec la technique d'initialisation proposée est nettement meilleur. En effet, par exemple, en observant l'image synthétique nous remarquons que, dans le cas de l'initialisation aléatoire, trois classes sont confondues : ce sont les deux classes circulaires et la classe de forme aléatoire. Le résultat est meilleur dans le cas de l'initialisation proposée car la classe de forme aléatoire est séparée des classes circulaires.



Fig. II.7 : Résultats de la segmentation vectorielle par l'algorithme EM en utilisant une initialisation aléatoire et l'initialisation proposée  
 a) Images originales b) Résultats de la segmentation par EM en utilisant une initialisation aléatoire c) Résultats de la segmentation par EM en utilisant l'initialisation proposée

#### 4.4 Résultats de la segmentation des images multi-composantes par l'algorithme EM

Dans cette section, nous allons présenter les résultats de la segmentation des images couleur (Fig. II.8) et Meteosat (Fig. II.9) en utilisant l'algorithme EM.

Cinq images couleur, dont quatre sont réelles et une est synthétique, sont testées. La figure II.8 présente ces images couleurs originales sur la première colonne. Les résultats de la segmentation vectorielle sont présentés sur la deuxième colonne et ceux de la segmentation marginale sont donnés sur la dernière colonne. L'affichage des résultats de la segmentation se fait avec la couleur moyenne de chaque classe de l'image dans le cas où cet affichage permet la visualisation de toutes les régions détectées. Dans le cas contraire, nous donnons un affichage avec de fausses couleurs.



Fig. II.8 : Résultats de la segmentation par l'algorithme EM des images couleur  
 1) Image "Synthétique" 2) Image "Avion" 3) Image "Fleurs" 4) Image "Peppers" 5) Image "Jelly"  
 a) Images originales b) résultats de la segmentation vectorielle c) Résultats de la segmentation marginale

L'image Meteosat que nous utilisons décrit la région nord-ouest de l'Afrique englobant le territoire algérien. Elle présente des régions nuageuses et d'autres régions non nuageuses. Sur la première ligne de la figure II.9, sont représentées en composition colorée les cinq composantes infrarouges de cette image. La dernière ligne fournit les résultats de la segmentation de cette image en utilisant l'algorithme EM de manière vectorielle et marginale.

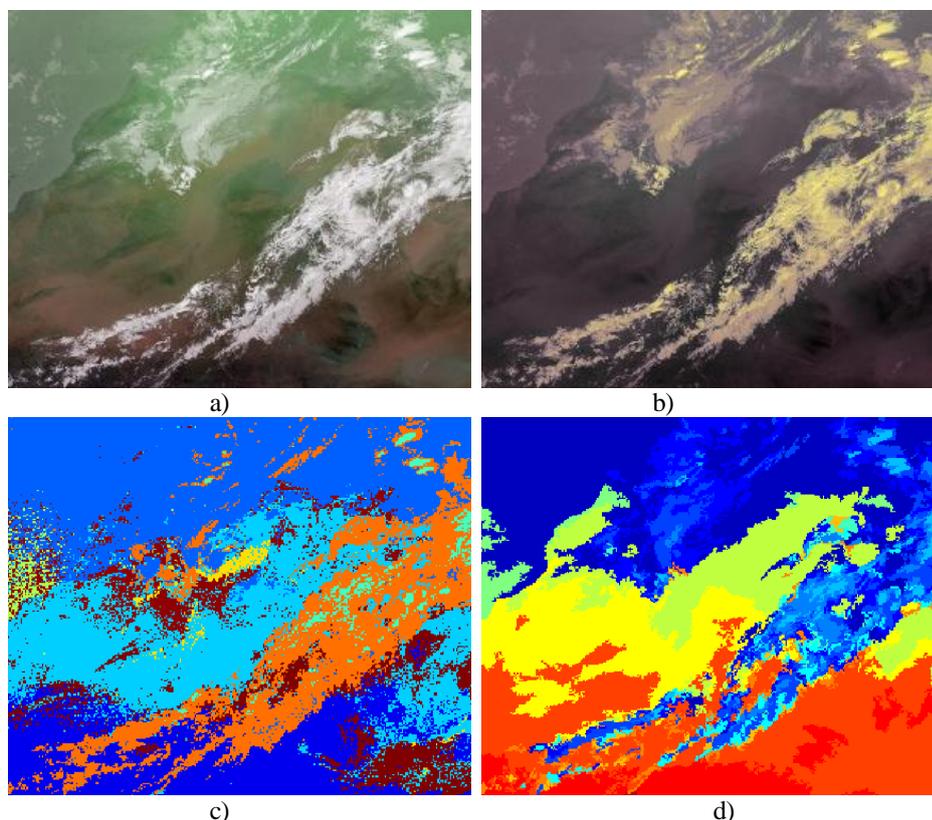


Fig. II.9 : Résultats de la segmentation par l'algorithme EM de l'image Meteosat prise le 15/12/2006 à 12h00mn en 15 classes

- a) Composition colorée des composantes IRN8.7, IRN9.7 et IRN10.8 b) Composition colorée des composantes IRN12.0, IRN13.4 et  $(\text{IRN12.0} + \text{IRN13.4})/2$  ; c) résultats de la segmentation vectorielle d) Résultats de la segmentation marginale

## 5 Segmentation des images multi-composantes en utilisant l'algorithme K-means

### 5.1 Principe de l'algorithme K-means

L'algorithme K-means est un des algorithmes les plus connus dans la communauté de classification des données. Dans cet algorithme, chaque classe est représentée par la moyenne (mean) ou la moyenne pondérée de ses points de données constituant ainsi son centre. K-means est un algorithme itératif. Il commence avec un ensemble de  $K$  points de référence (centres) choisis de manière aléatoire ou par l'utilisateur. Au début, les points de données sont partitionnés en  $K$  classes : un point appartient à une classe si le point de référence de cette classe est le plus proche de lui. La mise à jour des points de référence et l'affectation des points de données aux classes sont réalisées pendant les itérations successives.

Il y a plusieurs versions de K-means [52] qui diffèrent dans la mise à jour des classes et/ou dans le critère pour faire cette mise à jour. Les plus connus sont les algorithmes Standard K-means, K-means de Lloyd et Continuous K-means [53].

Dans l'algorithme dit Standard K-means, la mise à jour des centres est faite pendant et après une itération. Si un point appartient à une classe et que pour lui, le centre de cette classe est le point de référence le plus proche, alors il n'y aura aucun ajustement. Mais si après avoir affecté un point  $y$  à une classe  $C_i$ , on trouve qu'il y a une autre classe  $C_j$  dont le centre est le point de référence le plus proche de  $y$  que celui de  $C_i$ , alors il faut réaffecter  $y$  à la classe  $C_j$  et recalculer les centres des deux classes. Les points de référence de ces deux classes se déplacent donc vers de nouveaux centres. Dans l'algorithme de Lloyd, les ajustements sont réalisés en calculant le centre pour chaque classe et en utilisant ces centres comme les points de référence dans l'itération suivante pour tous les points de données. La mise à jour des centres n'est faite qu'après une itération. Dans l'algorithme Continuous K-means proposé par McQueen [54] les  $K$  centres initiaux sont choisis arbitrairement parmi les points de la population entière. Si l'échantillon est assez gros, alors la distribution des points de référence initiaux pourrait refléter celle des points de la population. Dans cet algorithme, contrairement au standard K-means où tous les points sont séquentiellement examinés, on n'examine qu'un échantillon aléatoire des points. Si le jeu de données est gros et l'échantillon est représentatif du jeu de données, alors l'algorithme peut converger plus vite qu'un algorithme qui doit examiner séquentiellement tous les points [53].

Dans ce travail, nous avons opté pour l'algorithme standard des K-means, qui dans sa version la plus simple nécessite la connaissance préalable du nombre  $K$  de classes. Son principe consiste à choisir initialement  $K$  centres définissant ainsi une partition  $C = \{C_1, C_2, \dots, C_K\}$  de  $K$  classes. Chacune des observations à classer est assignée à l'une des  $C_k$  classes selon une mesure de similarité entre chaque centre des classes et l'observation considérée. Une nouvelle partition  $C$  est ainsi obtenue et permet de redéfinir les nouveaux centres des classes. Ce processus itératif, prend fin lorsqu'un critère d'arrêt est satisfait. La méthode des K-means, également connue sous le nom de procédure de minimisation de l'erreur quadratique [54], fonctionne suivant l'algorithme de la figure II.10. Il s'agit de classer un ensemble de données  $Y = (y_1, y_2, \dots, y_n)$  avec  $y_i \in \mathbb{R}^D$  en  $K$  classes où chaque classe  $C_k$  est définie par son centre  $\bar{y}_k = (\bar{y}_{k1}, \bar{y}_{k2}, \dots, \bar{y}_{kp})$ , avec :

$$\bar{y}_{kj} = \frac{1}{N_k} \sum_{y_i \in C_k} y_{ij} \quad (\text{II.14})$$

$j$  étant l'indice de l'attribut,  $N_k$  le nombre d'observations de la classe  $k$  et  $y_{ij}$  l'attribut  $j$  de l'individu  $y_i$ . Chaque individu  $y_i$  se trouvant dans la classe  $k$  est individuellement réassigné à la classe dont la distance euclidienne par rapport à son centre est la plus faible. Les distances de l'individu  $y_i$  par rapport aux différentes classes sont définies par :

$$d_k(y_i) = d(y_i, \bar{y}_k) = \|y_i - \bar{y}_k\|^2 \quad (\text{II.15})$$

Le critère à optimiser est défini par :

$$J(C) = \sum_{k=1}^K \sum_{y_i \in C_k} d(y_i, \bar{y}_k)^2 = \sum_{k=1}^K J_k(C) \quad (\text{II.16})$$

La partition  $C^*$  qui optimise ce critère est définie par :  $J(C^*) = \min_C J(C)$

1. Donner le nombre  $K$  de classes, ainsi qu'une partition initiale  $C$  des  $n$  observations de l'échantillon en  $K$  classes.
2. Calculer  $J(C)$  et  $\bar{y}_k$ ,  $k = 1, 2, \dots, K$ .
3. Pour chaque observation  $y_i$  affectée initialement à la classe  $C_k$ ,  $i$  variant de 1 à  $n$  :
  - Calculer  $d_k(y_i)$ ,  $\forall k = 1, 2, \dots, K$ .
  - Affecter  $y_i$  à la classe  $C_s$  telle que :  $d_s(y_i) \leq d_k(y_i)$ ,  $\forall k = 1, 2, \dots, K$ .
  - Calculer  $\bar{y}_k$  et  $\bar{y}_s$ .
  - Recalculer  $J(C)$ .
4. Si  $J(C)$  a changé durant l'étape 3, revenir en 3, sinon arrêter.

Fig. II.10 : Algorithme K-means

## 5.2 Résultats de la segmentation des images multi-composantes en utilisant l'algorithme K-means

K-means est un algorithme très populaire et beaucoup utilisé grâce à sa simplicité. Il n'implique pas de longs calculs et converge rapidement. Par contre, il impose de donner le nombre de classes désirées et les partitions de départ au début de l'algorithme. Les performances de cette procédure dépendent fortement des conditions initiales. En ce qui nous concerne nous utiliserons les mêmes paramètres initiaux que ceux de l'algorithme EM afin de réaliser une comparaison objective des différentes méthodes. En effet, il n'est pas évident de comparer les résultats obtenus avec des méthodes de classification différentes si l'initialisation est réalisée en utilisant des paramètres initiaux différents. Les algorithmes convergeraient éventuellement vers des optimums locaux différents. Pour lever cette difficulté, nous avons choisie d'utiliser les mêmes paramètres initiaux pour tous les algorithmes qui nécessitent une initialisation. Les résultats de la segmentation en utilisant l'algorithme K-means des mêmes images couleur et Meteosat que celles utilisées précédemment sont fournis en figures II.11 et II.12.

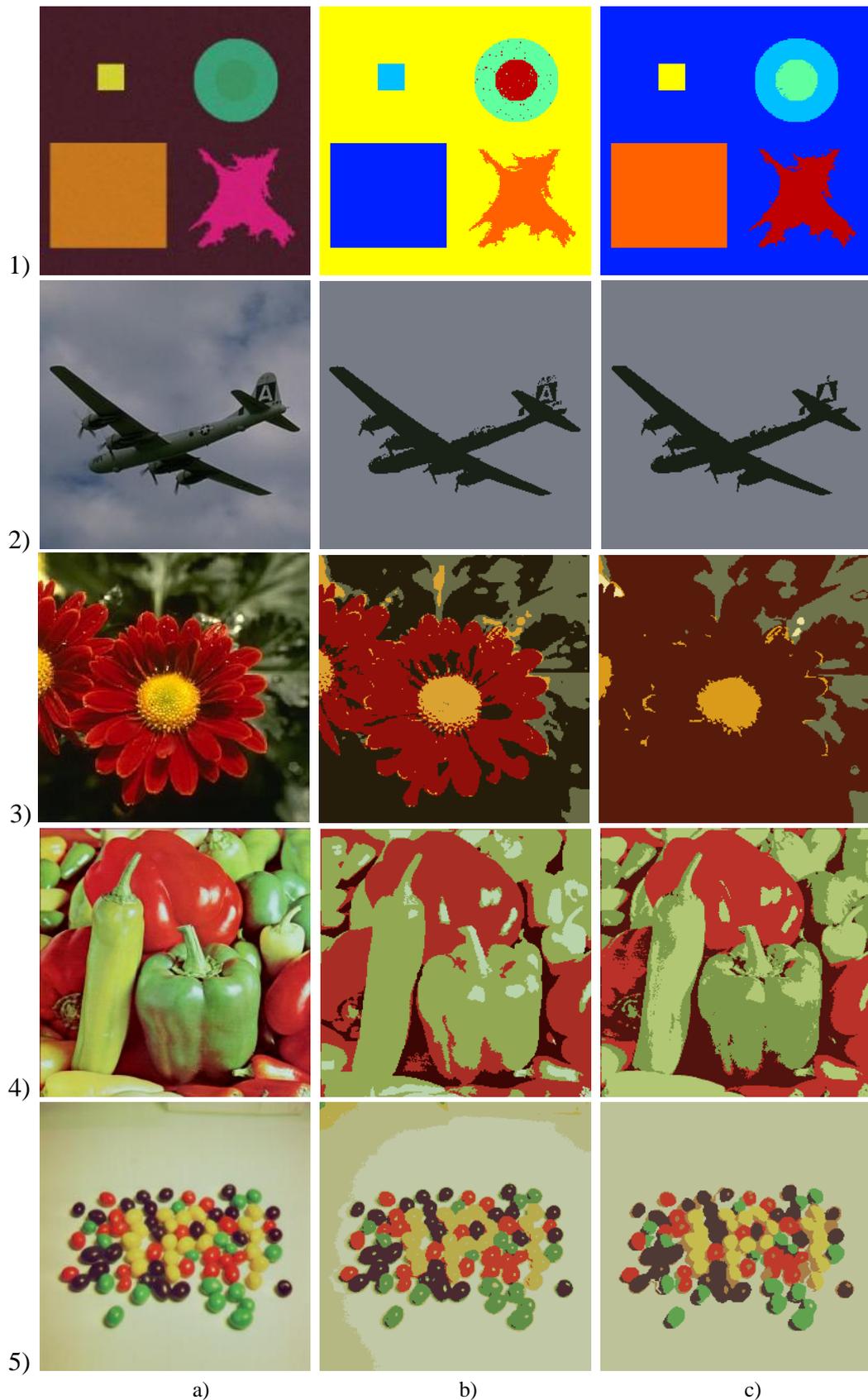


Fig. II.11: Résultats de la segmentation par l’algorithme K-means des images couleur  
 1) Image "Synthétique" 2) Image "Avion" 3) Image "Fleurs" 4) Image "Peppers" 5) Image "Jelly"  
 a) Images originales b) résultats de la segmentation vectorielle c) Résultats de la segmentation marginale

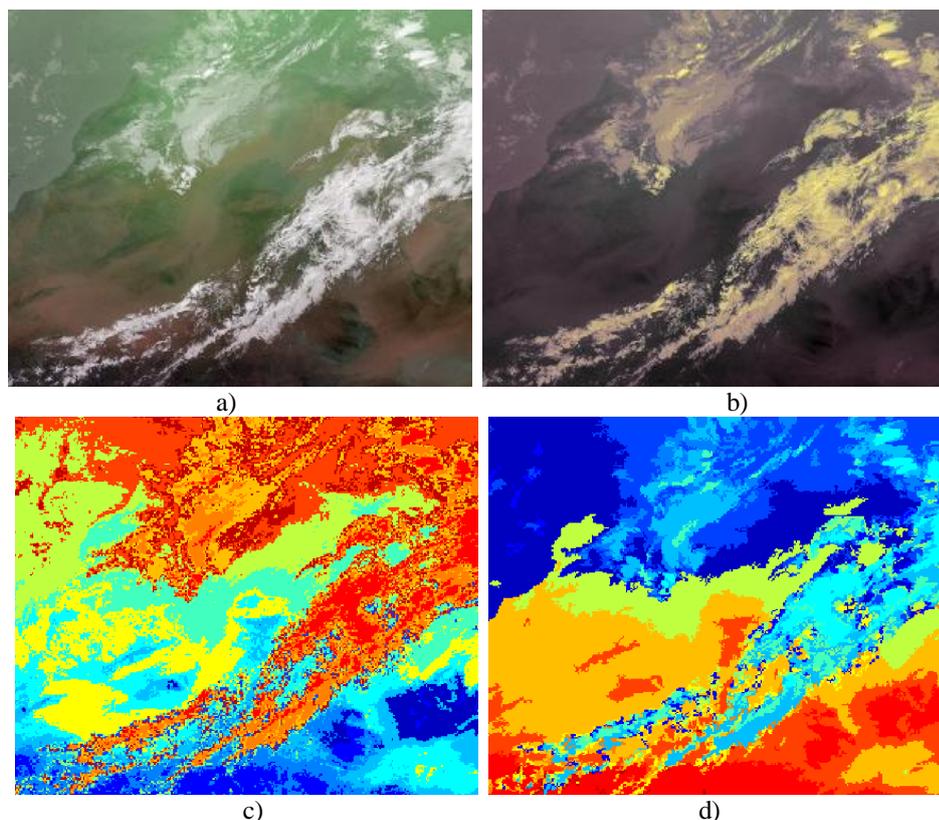


Fig. II.12 : Résultats de la segmentation par l'algorithme K-means de l'image Meteosat prise le 15/12/2006 à 12h00mn en 15 classes

- a) Composition colorée des composantes IRN8.7, IRN9.7 et IRN10.8 b) Composition colorée des composantes IRN12.0, IRN13.4 et  $(\text{IRN12.0} + \text{IRN13.4})/2$ ;  
 c) résultats de la segmentation vectorielle d) Résultats de la segmentation marginale

## 6 Segmentation des images multi-composantes en utilisant l'algorithme FCM

### 6.1 Principe de l'algorithme FCM

C-moyennes floues ou Fuzzy C-Means (FCM) est un algorithme de classification non-supervisée floue. Ruspini fut le premier à proposer, en 1969, une approche de classification combinant le concept de sous-ensemble flou et les techniques basées sur la minimisation d'un critère [55, 56]. Un peu plus tard, Dunn [57] et surtout Bezdek [58, 59] ont largement généralisé l'approche de Ruspini.

L'idée de base est qu'un individu peut appartenir à plusieurs classes avec des degrés d'appartenance différents, contrairement à la classification exclusive où un individu ne peut appartenir qu'à une seule classe. Les degrés d'appartenance des observations  $Y=(y_1, y_2, \dots, y_n)$  aux différentes classes  $C=(C_1, C_2, \dots, C_K)$  sont présentés dans une matrice  $U(\mu_{ik})$  avec  $i=1, \dots, n$  ( $n$  est le nombre d'observations) et  $k=1, \dots, K$  ( $K$  est le nombre de classes).

Ces degrés d'appartenance doivent vérifier les conditions suivantes :

- Tous les degrés d'appartenance appartiennent à l'intervalle  $[0,1]$ .

- La somme de tous les degrés d'appartenance d'une donnée quelconque  $y_i$  aux différentes classes  $C_k$  est égale à 1.
- Aucun sous-ensemble flou ne doit être vide, par conséquent, la somme de tous les degrés d'appartenance d'une même classe doit être strictement supérieure à 0.
- Aucun sous-ensemble flou ne peut être l'ensemble  $Y$  lui même, donc la somme des degrés d'appartenances d'une même classe doit être strictement inférieure à  $n$  :

Ces conditions sont donc formalisées par :

$$\left\{ \begin{array}{l} \mu_{ik} \in [0,1] \quad \forall i \in \{1,2,\dots,n\}, \forall k \in \{1,2,\dots,K\} \\ \sum_{k=1}^K \mu_{ik} = 1 \quad \forall i \in \{1,2,\dots,n\} \\ 0 < \sum_{i=1}^n \mu_{ik} < n \quad \forall k \in \{1,2,\dots,K\} \end{array} \right. \quad (\text{II.17})$$

L'algorithme des C-moyennes floues effectue une optimisation itérative en évaluant de façon approximative les minimums d'une fonction d'erreur. Il existe toute une famille de fonctions d'erreur associées à cet algorithme qui se distinguent par des valeurs différentes prises par un paramètre réglable,  $m$ , appelé indice de flou (*fuzzy index*) et qui détermine le degré de flou de la partition obtenue.

La fonction objective à minimiser dans le cas de la méthode FCM est :

$$J_m(C) = \sum_{i=1}^n \sum_{k=1}^K \mu_{ik}^m d_{ik}^2 \quad (\text{II.18})$$

où  $1 \leq m < \infty$  et  $d_{ik}$  est la distance séparant l'objet  $y_i$  du centre  $\bar{y}_k$  de la classe  $k$ . Cette distance exprime la similarité entre l'observation et le centre d'une classe telle que :

$$d_{ik} = \|y_i - \bar{y}_k\| \quad (\text{II.19})$$

Le carré de la distance séparant un vecteur  $y_i$  d'un centre  $\bar{y}_k$ ,  $(d_{ik})^2$ , est pondéré par la puissance  $m^{\text{ème}}$  du degré d'appartenance de la donnée  $y_i$  à la classe  $k$  ( $\mu_{ik}^m$ ).  $J_m$  est donc une erreur quadratique généralisée et sa minimisation conduit, théoriquement à la partition optimale. Comme pour la méthode K-means, la méthode FCM fonctionne d'une manière itérative. A chaque itération, est effectuée une mise à jour du degré d'appartenance  $\mu_{ik}$  et des centres  $\bar{y}_k$  des classes. La mise à jour des centres des classes est donnée par l'expression suivante :

$$\bar{y}_k = \frac{\sum_{i=1}^n \mu_{ik}^m y_i}{\sum_{i=1}^n \mu_{ik}^m} \quad (\text{II.20})$$

La mise à jour des centres des classes modifie la répartition des observations, par conséquent une mise à jour de la matrice d'appartenance est nécessaire. Elle est effectuée en utilisant l'expression suivante :

$$\mu_{ik} = \begin{cases} 1 & \text{si } d_{ik} = 0 \\ 0 & \text{si } d_{il} = 0 \text{ pour } l \neq k \\ \frac{1}{\sum_{l=1}^K \left(\frac{d_{ik}}{d_{il}}\right)^{2/m-1}} & \text{sinon} \end{cases} \quad (\text{II.21})$$

La figure II.13 présente l'algorithme Fuzzy C-means où la variable  $t$  indique le numéro de l'itération. Le critère d'arrêt est indiqué par la dernière étape de l'algorithme. Il consiste à vérifier si la répartition entre deux itérations est la même ou non.  $\varepsilon$  est un nombre réel dont la valeur est comprise entre 0 et 1.

1. Fixer le nombre  $K$  de classes et le facteur flou  $m$ .
2. Initialisation des centres des classes,  $t=0$ .
3. Déterminer la matrice d'appartenance  $U^{(0)}$  en utilisant les équations II.19 et II.20.
4.  $t=t+1$  et déterminer les nouveaux centres des classes en utilisant l'équation II.20.
5. Déterminer la matrice d'appartenance  $U^{(t)}$  en utilisant les équations II.19 et II.21.
6. Si  $\|U^{(t+1)} - U^{(t)}\| < \varepsilon$  s'arrêter; sinon aller à l'étape 4.

Fig. II.13 : Algorithme Fuzzy C-means

## 6.2 Résultats de la segmentation des images multi-composantes en utilisant l'algorithme FCM

L'algorithme FCM nécessite la connaissance préalable du nombre de classes  $K$ . Il est aussi très sensible à la répartition initiale. Nous utilisons les mêmes paramètres initiaux que ceux des l'algorithme EM et K-means. Nous reprenons donc les mêmes vecteurs donnant les centres des classes et nous utiliserons ces centres pour calculer la matrice des degrés d'appartenance en utilisant les formules II.21. Les résultats de la segmentation en utilisant l'algorithme FCM, comme dans les deux cas précédents sont fournis dans les figures II.14 et II.15.

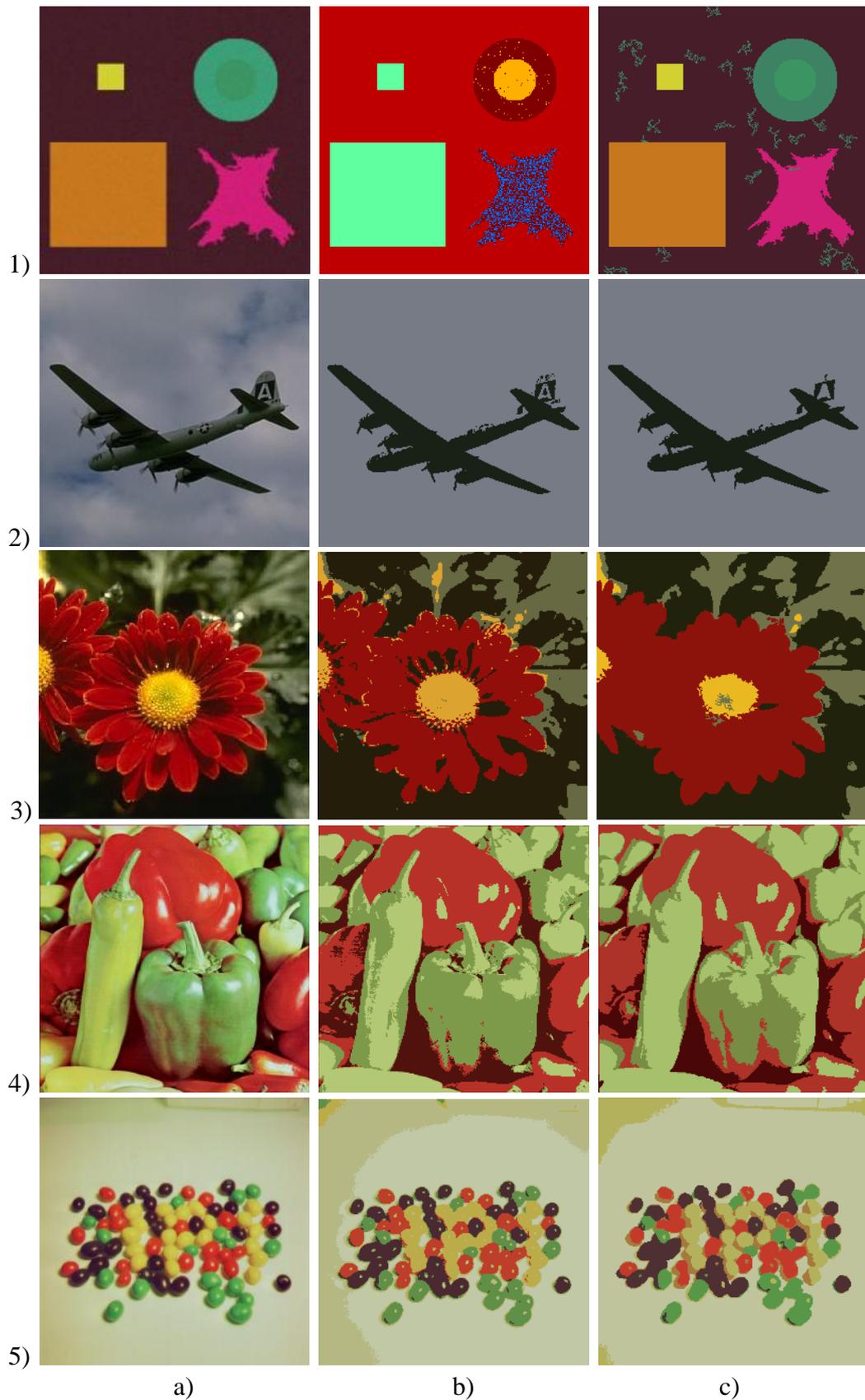


Fig. II.14 : Résultats de la segmentation par l'algorithme FCM des images couleur  
 1) Image "Synthétique" 2) Image "Avion" 3) Image "Fleurs" 4) Image "Peppers" 5) Image "Jelly"  
 a) Images originales b) résultats de la segmentation vectorielle c) Résultats de la segmentation marginale

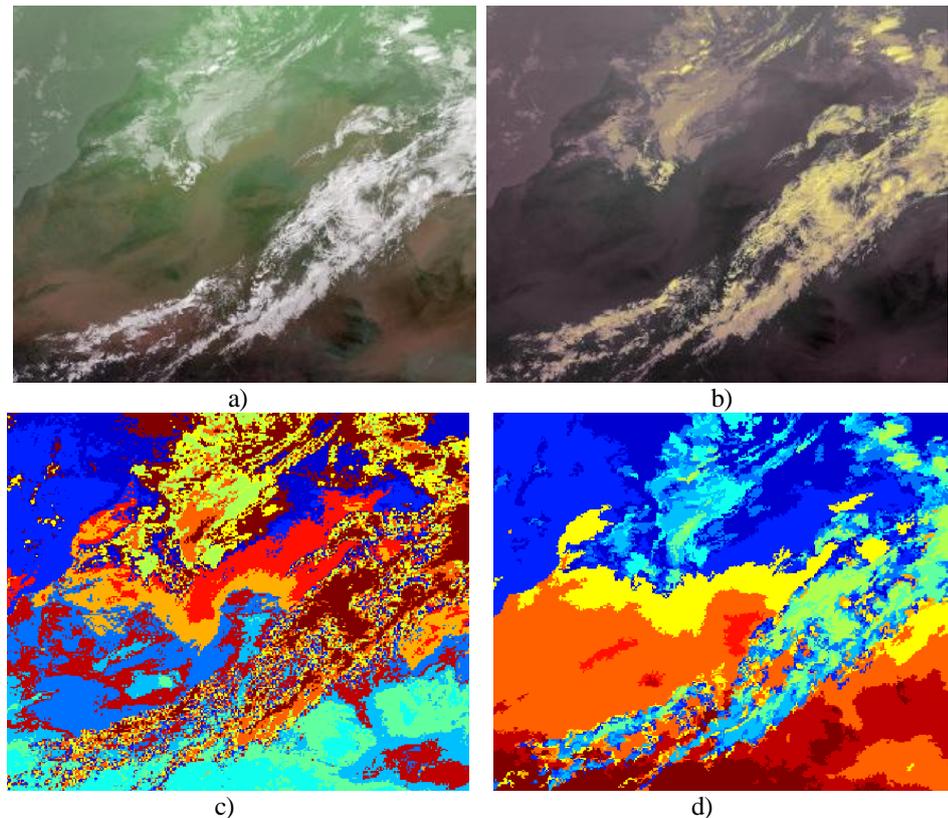


Fig. II.15 : Résultats de la segmentation par l'algorithme FCM de l'image Meteosat prise le 15/12/2006 à 12h00mn en 15 classes  
 a) Composition colorée des composantes IRN8.7, IRN9.7 et IRN10.8 b) Composition colorée des composantes IRN12.0, IRN13.4 et (IRN12.0+IRN13.4)/2; c) résultats de la segmentation vectorielle d) Résultats de la segmentation marginale

## 7 Segmentation des images multi-composantes en utilisant les cartes auto-organisatrice de Kohonen

### 7.1 Principe des cartes auto-organisatrices

Les cartes topologiques ou cartes auto-organisatrices (*Self Organization Maps*) (SOM) sont des réseaux de neurones à apprentissage non supervisé. Elles ont été introduites pour la première fois par T. Kohonen en 1981 [60, 61]. La particularité la plus importante de ces cartes est qu'elles rendent possible l'auto-organisation des observations en groupes. Avant de développer le principe des cartes auto-organisatrices, il nous semble nécessaire d'introduire les différentes notations utilisées.

Soit  $\mathcal{Y}$  l'ensemble qui représente l'espace des observations de dimension  $D$  tel que  $\mathcal{Y} \subset \mathbb{R}^D$ . Soit le sous-ensemble  $Y = \{y_i ; i = 1, \dots, n\}$  de  $\mathcal{Y}$ , constitué de  $n$  individus qui formera l'ensemble d'apprentissage permettant d'estimer les différents paramètres du modèle. En ce qui nous concerne,  $Y$  représente l'image multi-composante constitué des pixels  $y_i$  qui sont de

dimension  $D=3$  pour l'image couleur traduisant les trois composantes colorimétriques et  $D=5$  pour les image Meteosat donnant les composantes spectrales.

Soit l'ensemble  $W = \{w_c; c = 1, \dots, p\}$  de  $p$  vecteurs de  $\mathcal{Y}$ . Ces vecteurs de dimension  $D$  constituent l'ensemble des référents.

On peut représenter l'ensemble constitué par la carte et les référents sous la forme d'un réseau de neurones (Fig. II.16) constitué de deux couches :

- une couche d'entrée qui sert à la présentation des observations à classer ; les états de tous ces neurones sont forcés aux valeurs des observations
- et une couche d'adaptation 2D notée  $\varphi$  qui est formée du treillis des neurones qui forment la carte ; les neurones utilisés à ce niveau sont des simples neurones "distances", chacun d'entre eux étant connecté à tous les éléments de la couche d'entrée.

Soit  $\mathcal{F}$  une fonction d'affectation qui est une application de  $\mathcal{Y}$  dans l'ensemble des indices  $\{1, \dots, p\}$ . Cette fonction  $\mathcal{F}$  permet d'assigner à chaque vecteur d'entrée  $y \in Y$ , un neurone d'indice  $c \in \varphi$  choisi parmi  $p$  désigné par son vecteur référent  $w_c$ , tel que le vecteur référent  $w_c$  est le plus proche de  $y$ .

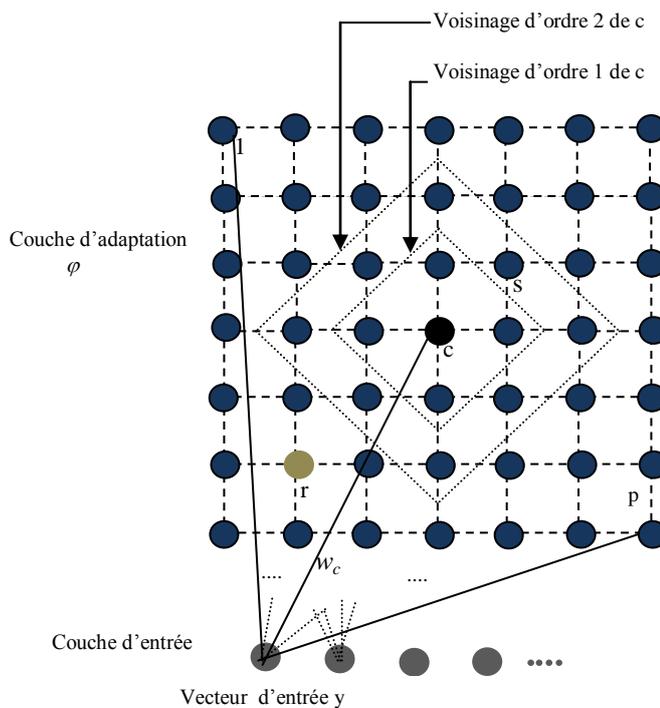


Fig. II.16 : Représentation d'une carte topologique à 2D de Kohonen

Il existe entre les neurones émetteurs et récepteurs un mécanisme d'interaction latérale qui dépend de la distance entre les neurones concernés [60, 61]. Ces cartes sont utilisées pour cartographier un espace réel, c'est-à-dire pour étudier la répartition de données dans un espace à grande dimension. En pratique, cette cartographie peut servir à réaliser des tâches de discrétisation, quantification vectorielle, ou classification et regroupement.

La carte topologique peut être considérée comme un graphe dont la structure induit une distance  $\delta$  sur  $\varphi$  pour toute paire de neurones  $(c,r)$  de cette carte.  $\delta(c,r)$  est définie comme étant la longueur du plus court chemin entre  $c$  et  $r$  sur le graphe  $\varphi$ . Dans la figure II.16 les distances entre  $c$  et  $r$ , et  $c$  et  $s$  sont respectivement  $\delta(c,r)=4$  et  $\delta(c,s)=2$ .

Les distances  $\delta(c,r)$  qui lient le neurone  $c$  aux autres neurones  $r$  de la carte permettent de faire varier l'influence relative des différents neurones. Cette importance est quantifiée par la fonction de voisinage  $V(\delta(c,r))$  qui force les neurones qui se trouvent dans le voisinage de  $c$  à rapprocher leurs vecteurs référents du vecteur d'entrée  $y$ . Moins un neurone est proche du vainqueur dans la grille, moins son déplacement est important. Les fonctions de voisinage sont généralement paramétrées par un facteur température  $T$  afin de tenir compte de la taille du voisinage. Les plus utilisées dans la pratique sont données par les équations ci-dessous et elles sont représentées sur la figure II.17. Dans tous les cas la fonction de voisinage  $V^T(\delta(c,r))$  décroît lorsque la distance  $\delta(c,r)$  croît.

$$V^T(\delta) = \begin{cases} 1 & \text{si } |\delta| < T \\ 0 & \text{sinon} \end{cases} \quad (\text{II.22})$$

$$V^T(\delta) = \exp\left(-\frac{|\delta|}{T}\right) \quad (\text{II.23})$$

$$V^T(\delta) = \exp\left(-\frac{\delta^2}{T^2}\right) \quad (\text{II.24})$$

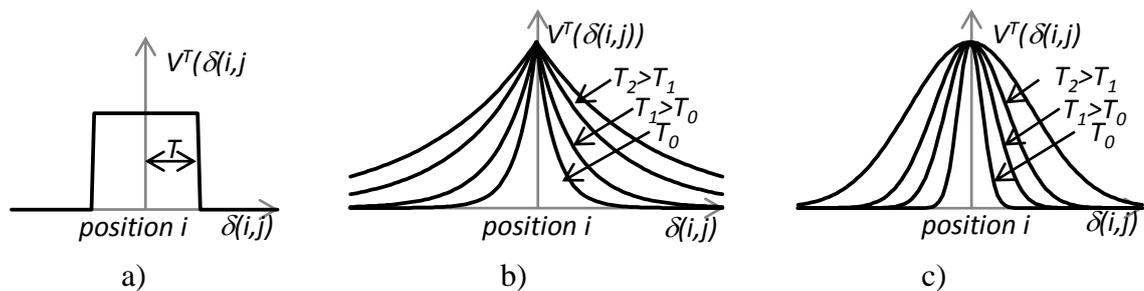


Fig. II.17 : Exemples de fonctions de voisinage

- a) fonction de voisinage à seuil équation (II.22) b) fonction de voisinage de type exponentiel équation (II.23)  
c) fonction de voisinage de type gaussien équation (II.24)

Les algorithmes des cartes auto-organisatrices minimisent une fonction de coût notée  $J_{som}^T$  [62]. Le minimum de cette fonction fournit une partition formée de sous-ensembles qui sont suffisamment compacts, mais pour lesquels on est capable de définir un ordre induit à partir de la topologie de la carte. La fonction coût est donnée par la formule suivante :

$$J_{som}^T(\mathcal{F}, W) = \sum_{y_i \in Y} \sum_{c \in C} V^T(\delta(c, \mathcal{A}(y_i))) \|y_i - w_c\|^2 \quad (\text{II.25})$$

La minimisation de cette fonction pour une valeur  $T$  fixée, est réalisée par itérations successives chacune se décomposant en deux phases. La première phase affecte l'ensemble des observations et la seconde minimise la valeur de la fonction coût associée à la partition. La phase d'affectation minimise la fonction  $J_{som}^T(\mathcal{F}, W)$  par rapport à la fonction d'affectation. On suppose que l'ensemble  $W$  des référents est constant et égale à la valeur calculée précédemment. Ainsi, chaque observation  $y$  est affectée au référent le plus proche au sens d'une distance. La fonction d'affectation utilisée par Kohonen est donnée par :

$$\mathcal{A}(y_i) = \arg \min_c \|y_i - w_c\|^2 \quad (\text{II.26})$$

Durant la phase de minimisation, on minimise la fonction  $J_{som}^T(\mathcal{F}, W)$  par rapport à l'ensemble des référents  $W$ . Cette minimisation est effectuée en gardant la fonction d'affectation  $\mathcal{F}$  fixe et égale à la fonction calculée durant la phase précédente. La minimisation est donc obtenue pour la valeur qui annule la dérivée de  $J_{som}^T(\mathcal{F}, W)$ . Dans la version stochastique de Kohonen, il n'est pas obligatoire de trouver le minimum global de  $J_{som}^T(\mathcal{F}, W)$  pour  $\mathcal{F}$  fixée, il suffit de faire décroître sa valeur. Ainsi à l'itération  $t$  et pour un neurone  $c$  le référent est modifié selon la règle suivante :

$$w_c^t = w_c^{t-1} - \mu^t \frac{\partial J_{som}^T}{\partial w_c^{t-1}} \quad (\text{II.27})$$

Avec

$$\frac{\partial J_{som}^T}{\partial w_c} = 2 \sum_{y_i \in Y} V^T(\delta(c, \mathcal{A}(y_i))) (y_i - w_c) \quad (\text{II.28})$$

Cette méthode non adaptative suppose que l'on dispose de toutes les observations de l'ensemble d'apprentissage  $Y$ . La contribution d'une seule observation  $y_i$  à la correction de  $w_c$  est représentée par le terme de la somme  $2V^T(\delta(c, \mathcal{A}(y_i))) (y_i - w_c^{t-1})$ .

La règle de modification des référents prend alors la forme :

$$w_c^t = w_c^{t-1} - \mu^t V^T \left( \delta(c, \mathcal{A}(y_i)) \right) (w_c^{t-1} - y_i) \quad (\text{II.29})$$

où  $\mu^t$  représente le pas du gradient de l'itération  $t$ .  $\mu^t$  doit décroître avec les itérations. Au début de l'algorithme, la valeur de  $\mu^t$  est grande et la décroissance de la fonction  $J_{som}^T$  n'est pas strictement assurée. Par la suite, le pas du gradient devient suffisamment petit : la modification des référents à chaque itération est petite. Dans ce travail, nous avons testé trois types de fonctions pour le pas gradient qui sont :

$$\mu^t = \frac{1}{t} \quad (\text{II.30})$$

$$\mu^t = \frac{1}{\sqrt{t}} \quad (\text{II.31})$$

$$\mu^t = \frac{1}{t^2} \quad (\text{II.32})$$

L'algorithme de Kohonen est donné sur la figure II.18 [62].

**1. Phase d'initialisation**

- choisir la structure  $\varphi$  et la taille  $\tau$  de la carte ainsi que les  $p$  référents initiaux  $W^0$  (en général, d'une manière aléatoire).
- fixer les valeurs de  $T_{max}$  et  $T_{min}$  et le nombre d'itération  $N_{iter}$  ; prendre  $t=0$ .

**2. Etape itérative t**

- choisir une observation  $y_i$  (en général d'une manière aléatoire).
- calculer la nouvelle valeur de  $T$  en appliquant la formule :

$$T = T_{max} * \left( \frac{T_{min}}{T_{max}} \right)^{\frac{t}{N_{iter} - 1}} \quad (\text{II.33})$$

- Pour cette valeur du paramètre effectuer les deux phases suivantes :
  - phase d'affectation : affecter l'observation  $y_i$  au neurone  $\mathcal{A}(y_i)$  défini par la relation II.26
  - phase de minimisation : modifier les valeurs des référents  $W^t$  par la relation II.29.

**3.  $t=t+1$ , Répéter l'étape itérative jusqu'à ce que l'on atteigne  $t=N_{iter}$ .**

Fig. II.18 : Algorithme de l'auto-organisation des cartes de Kohonen

L'algorithme détermine le centre d'activité comme étant le voisinage de la carte associé au neurone dont l'état  $\|y_i - w_c\|$  est le plus petit. C'est l'utilisation de ce voisinage qui introduit les contraintes topologiques dans la représentation finale. Le paramètre  $T$  décroît au fur et à mesure des itérations. Lorsqu'il est petit, la modification de la relation II.29 ne concerne qu'un ensemble réduit de neurones.

## 7.2 Regroupement des référents

L'auto-organisation, telle que nous venons de la décrire, ne permet pas de résoudre des problèmes de classification. Chaque observation est affectée à un neurone de la carte, indépendamment de toute notion de classe. Des sous-ensembles cohérents, en relation avec l'application traitée, apparaissent donc à l'issue de l'apprentissage non supervisé. Si l'application envisagée consiste en un problème de classification en  $K$  classes, les différents sous-ensembles doivent s'intégrer au mieux avec celles-ci. On cherche alors à affecter chaque sous-ensemble à l'une de ces classes.

Puisque chaque sous-ensemble est associé à un neurone de la carte, le problème de classification se résume à celui de l'étiquetage de chaque neurone de la carte au moyen de l'une des  $K$  classes du problème.

Le nombre de neurones étant supérieur au nombre de classes, l'étiquetage peut se faire en regroupant au mieux ces neurones. L'étiquetage peut se faire de trois manières différentes :

- il est possible de demander à un expert du domaine d'application de reconnaître, grâce aux caractéristiques de cette observation, la classe à laquelle elle appartient,
- regrouper les neurones de la carte d'une manière statistique, le recours à l'expertise ne se faisant qu'à l'issue de cette phase,
- on peut aussi les regrouper en utilisant une classification hiérarchique ascendante.

Dans ce travail, c'est cette dernière stratégie qui sera utilisée. Il s'agit d'une méthode qui calcule une hiérarchie de partitions [63], chaque partition permet de regrouper d'une manière différente les neurones de la carte. Les différentes partitions de la hiérarchie sont déterminées d'une manière itérative, en commençant par la partition la plus fine qui est composée de l'ensemble des singletons (les neurones). La classification hiérarchique utilise cette partition initiale et procède à des regroupements successifs en fusionnant à chaque itération deux sous-groupes de neurones. Le choix des deux sous-groupes qui vont fusionner à une étape donnée est effectué à l'aide d'une mesure de similitude, définie entre deux sous-groupes. On choisit, parmi tous les couples de sous-groupes qui constituent la partition à cette étape, les deux sous-groupes de neurones les plus semblables, au sens de la mesure choisie. Pour ce qui nous concerne, la mesure que nous avons utilisée est une distance euclidienne radiométrique entre les sous-groupes de neurones voisins. En raison de la topologie de la carte, on ne regroupe que les neurones et les sous-groupes de neurones voisins dans la carte [64].

La classification hiérarchique permet d'engendrer un nombre variable de sous-groupes, car le processus de regroupement peut être arrêté à tout moment. Ainsi le nombre d'éléments de la partition que l'on choisie dépend du nombre  $K$  de classe recherchées.

L'algorithme de la classification ascendante hiérarchique (CAH) est donné sur la figure II.9.

1. Initialisation : considérer la partition formée par les neurones ; chaque neurone est alors affecté à un sous-groupe distinct. Choisir le nombre  $K$  de groupement que l'on souhaite former.
2. Pour une partition donnée, trouver les deux sous-groupes les plus proches au sens du critère de similitude choisi, et les fusionner de manière à former un seul sous-groupe.
3. Si le nombre de groupement de la partition courante est supérieur à  $K$ , revenir à l'étape 2, sinon l'algorithme se termine.

Fig. II.19 : L'algorithme de classification ascendante hiérarchique

### 7.3 Choix des paramètres ajustables

La méthode de segmentation précédemment décrite nécessite le choix de la fonction de voisinage  $V^T$ , le pas de gradient  $\mu_i$  et la taille de la carte topologique  $\tau$  [64].

Après l'auto-organisation de la carte de Kohonen, nous obtenons  $\tau$  référents finaux. S'en suit un regroupement de ces référents jusqu'à obtention du nombre de classes souhaité. Nous regroupons tous les référents adjacents (c'est-à-dire ayant une distance spatiale égale à 1 de manière hiérarchique ascendante.

La taille  $\tau$  de la carte topologique peut influencer sur le résultat de la segmentation des images. Pour mieux illustrer cette influence, nous considérons sur la figure II.20 les cas où  $\tau=4 \times 4$ ,  $\tau=7 \times 7$ ,  $\tau=10 \times 10$ ,  $\tau=15 \times 15$  et  $\tau=30 \times 30$ . Ainsi nous aurons respectivement une carte topologique de 16, 49, 100, 225 et 900 neurones. Nous donnons pour chaque résultat, le temps de calcul  $t_c$  obtenu en utilisant un PC portable HP Intel Core 2 duo de 1,8 GHz.

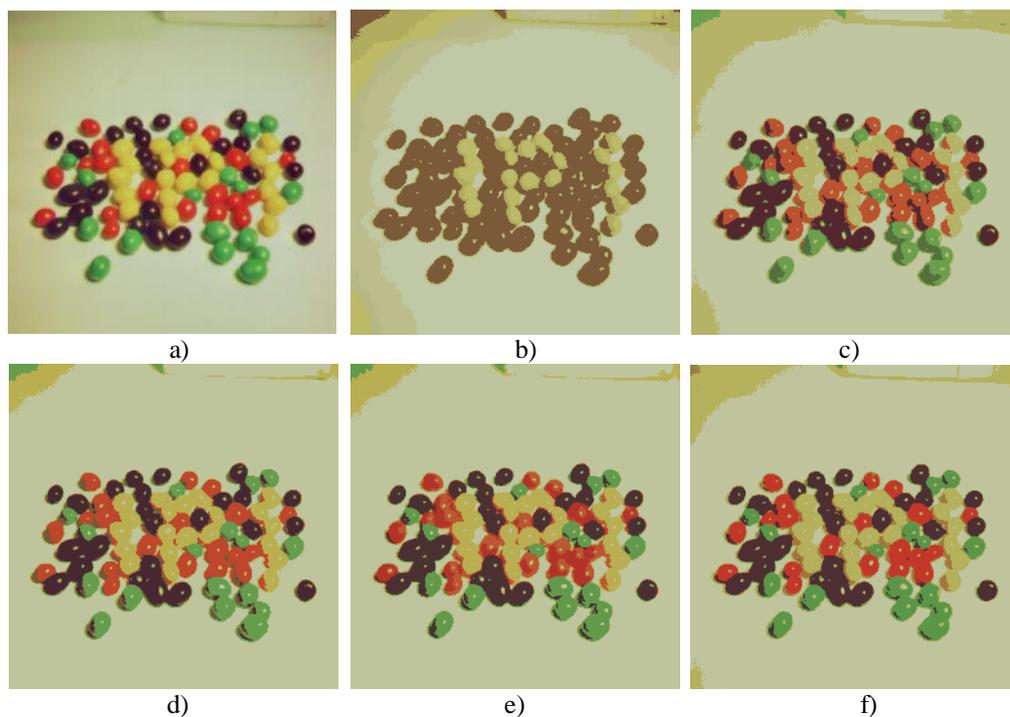


Fig. II.20 : Segmentation de l'image Jelly

a) Image initiale

Résultats de la classification en considérant :

b)  $\tau=4 \times 4$   $tc=17s$  c)  $\tau=7 \times 7$   $tc=29s$  d)  $\tau=10 \times 10$   $tc=49s$  e)  $\tau=15 \times 15$   $tc=95s$  f)  $\tau=30 \times 30$   $tc=388s$ 

Il est perceptible sur la figure II.20 que, plus la taille de la carte topologique augmente, meilleur est le résultat. Ce résultat est dû sans doute au fait que les différentes classes dans l'image seront mieux séparées lorsque le nombre de neurones est plus élevé. Nous pouvons augmenter ce nombre indéfiniment mais au détriment du temps de calcul. En effet pour une taille  $\tau=4 \times 4$  le temps de calcul est en moyenne de 17 secondes et pour  $\tau=30 \times 30$ , il dépasse les 6 minutes. Un compromis entre temps de calcul et qualité des résultats est donc nécessaire. En ce qui nous concerne, nous choisissons une taille  $\tau=10 \times 10$ , ce qui correspond à 100 neurones car les résultats sont appréciables et le temps de calcul n'atteint pas une minute. Au delà de cette valeur, nous augmentons le temps de calcul sans pour autant apporter de nettes améliorations aux résultats.

La fonction de voisinage  $V^T$  influe sur la modification des vecteurs référents dont la formule est donnée par l'équation II.29. Il est donc possible que le choix d'une fonction peut influencer sur le résultat final de la segmentation. Sur la figure II.21, nous affichons quelques résultats de la segmentation en considérant les trois exemples des fonctions de voisinages données par les équations II.22, II.23 et II.24.

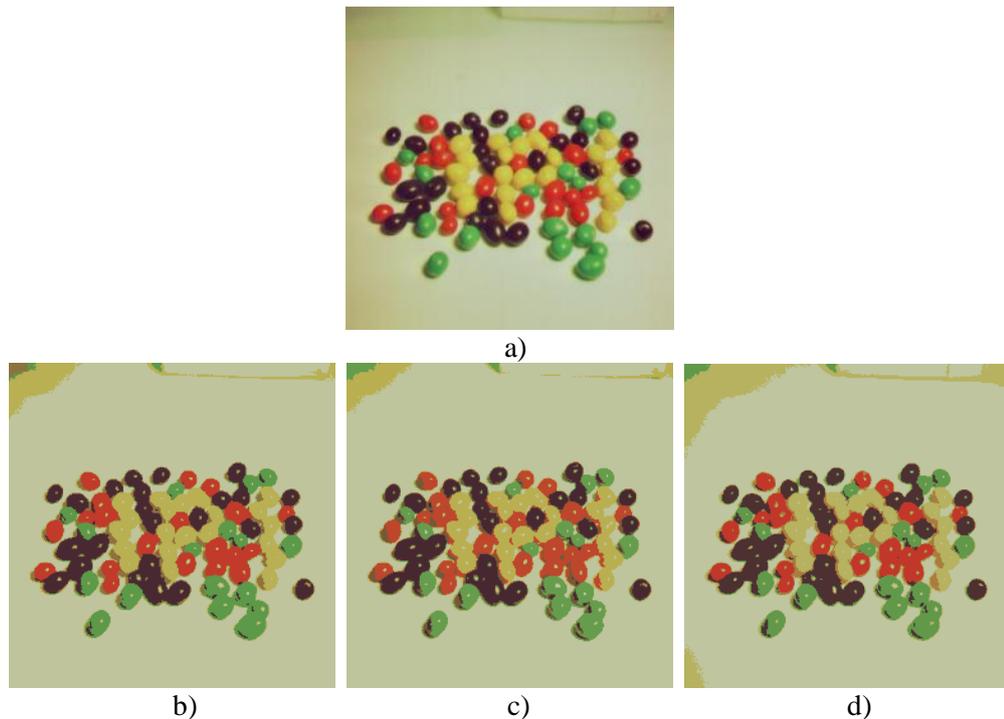


Fig. II.21 : Segmentation de l'image Jelly

a) Image initiale

Résultats de la classification en considérant la fonction

$$b) V^T(\delta) = \begin{cases} 1 & \text{si } |\delta| < T \\ 0 & \text{sinon} \end{cases} \quad c) V^T(\delta) = \exp\left(-\frac{|\delta|}{T}\right) \quad d) V^T(\delta) = \exp\left(-\frac{\delta^2}{T^2}\right)$$

Nous constatons que les résultats obtenus en faisant varier la fonction de voisinage sont assez semblables, il est alors assez difficile d'en choisir une. Dans ce travail nous optons pour la fonction  $V^T(\delta) = \exp\left(-\frac{|\delta|}{T}\right)$  tout en observant que les autres fonctions peuvent fournir d'aussi bons résultats.

Dans la formule de modification des vecteurs référents, le pas du gradient  $\mu^t$  est une fonction qui doit décroître en fonction des itérations. Ainsi si la valeur prise par ce pas est importante, l'influence de la valeur du pixel traité à l'itération en question peut modifier de manière importante le référent gagnant ainsi que ceux de son voisinage. Pour vérifier si le choix d'une fonction pas du gradient influe sur les résultats, nous considérons les trois fonctions données par les équations II.30, II.31 et II.32 appliquées sur l'image Jelly en figure II.22.

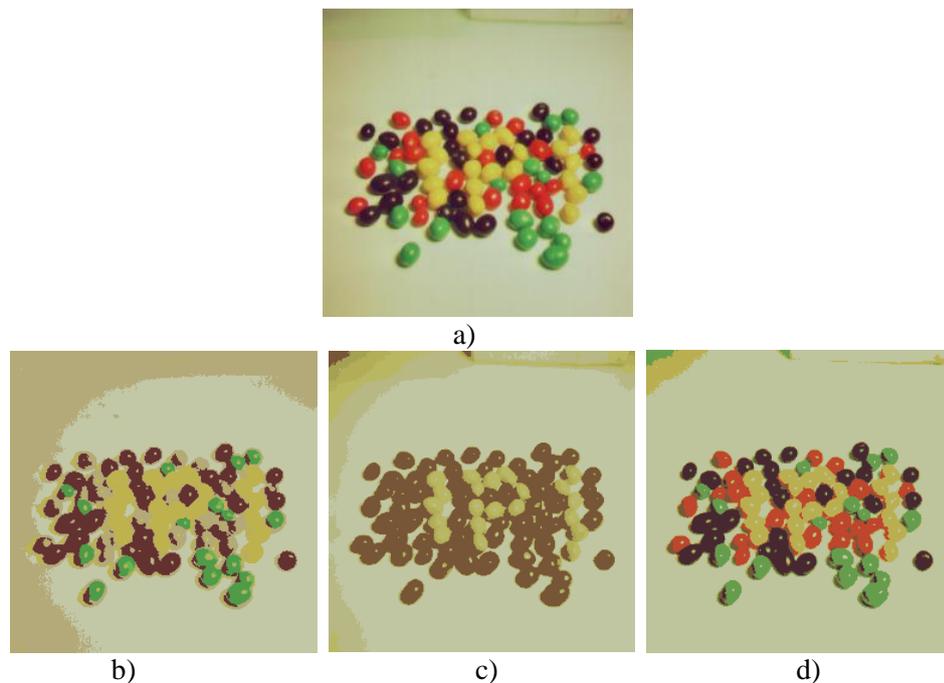


Fig. II.22 : Segmentation de l'image Jelly

a) Image initiale; Résultats de la classification en considérant b)  $\mu^t = \frac{1}{t^2}$  c)  $\mu^t = \frac{1}{t}$  d)  $\mu^t = \frac{1}{\sqrt{t}}$

D'après les résultats affichés sur cette figure, nous constatons que le choix de la fonction pas du gradient est primordiale. En effet, par exemple, pour la fonction  $\mu^t = \frac{1}{t^2}$ , les résultats sont mauvais, ils le sont moins pour la fonction  $\mu^t = \frac{1}{t}$  et ils sont nettement meilleurs pour la fonction  $\mu^t = \frac{1}{\sqrt{t}}$ . La seule exigence dans la littérature sur cette fonction est qu'elle doit décroître en fonction des itérations. Nous constatons que si cette décroissance est très rapide les résultats sont mauvais. C'est pour cette raison la que pour la suite de ce travail, nous considérons la fonction  $\mu^t = \frac{1}{\sqrt{t}}$ .

#### 7.4 Résultats de la segmentation des images multi-composantes par les cartes auto-organisatrice de Kohonen

L'algorithme de Kohonen ne nécessite que la connaissance préalable du nombre de classes  $K$ . l'initialisation de la carte des référents est réalisée de manière aléatoire. Pour différentes initialisations, les résultats sont assez semblables car la carte topologique se trouve au final organisée en fonction de cette initialisation. En effet les changements qui peuvent être observés sur la carte sont les positions dans celle-ci des groupements qui restent assez semblables.

Les résultats de la segmentation en utilisant l'algorithme des cartes auto-organisatrices de Kohonen sont fournis dans les figures II.23 et II.24.

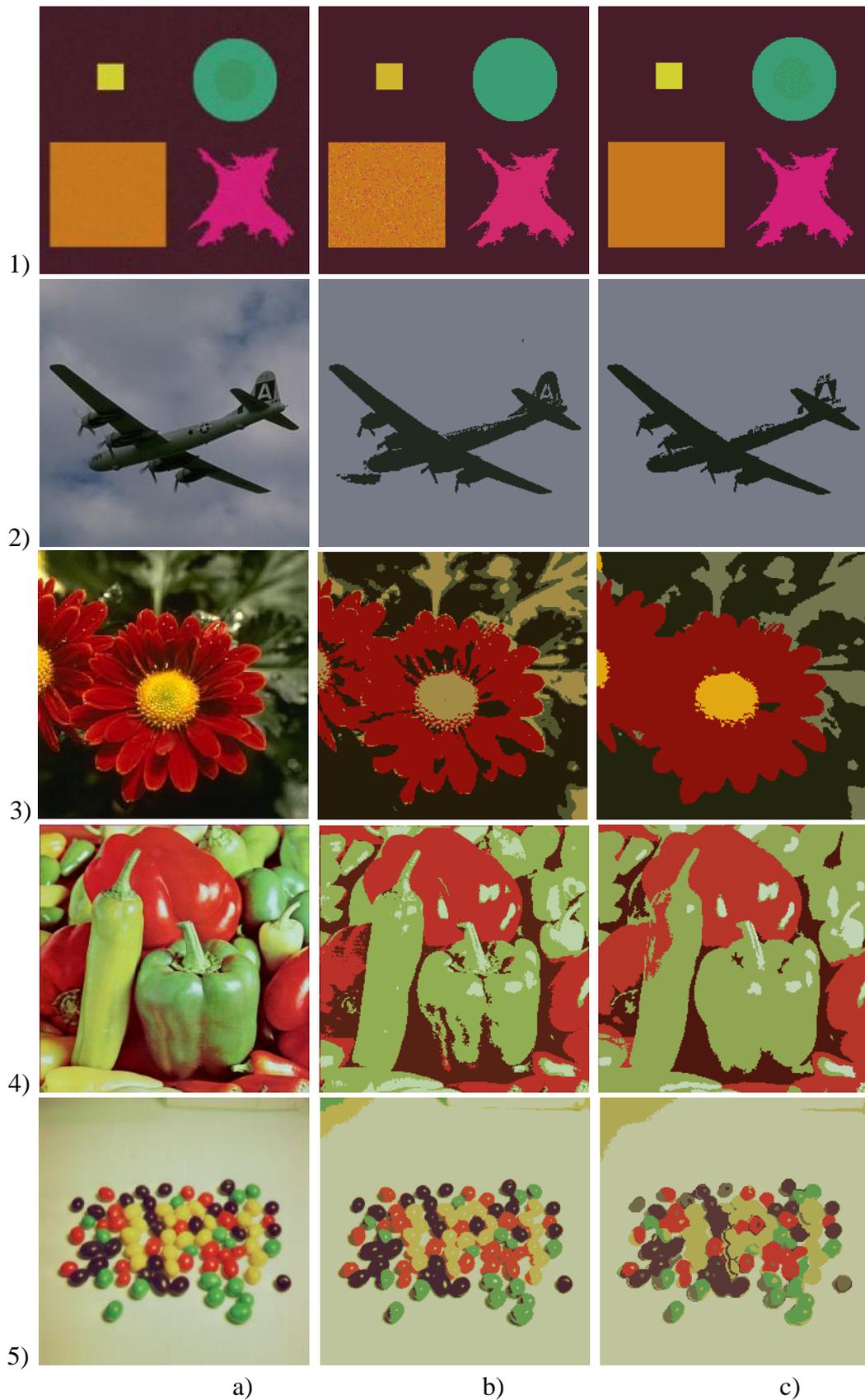


Fig. II.23 : Résultats de la segmentation par l'algorithme des cartes auto-organisatrices de Kohonen des images couleur

1) Image "Synthétique" 2) Image "Avion" 3) Image "Fleurs" 4) Image "Peppers" 5) Image "Jelly"  
 a) Images originales b) résultats de la segmentation vectorielle c) Résultats de la segmentation marginale

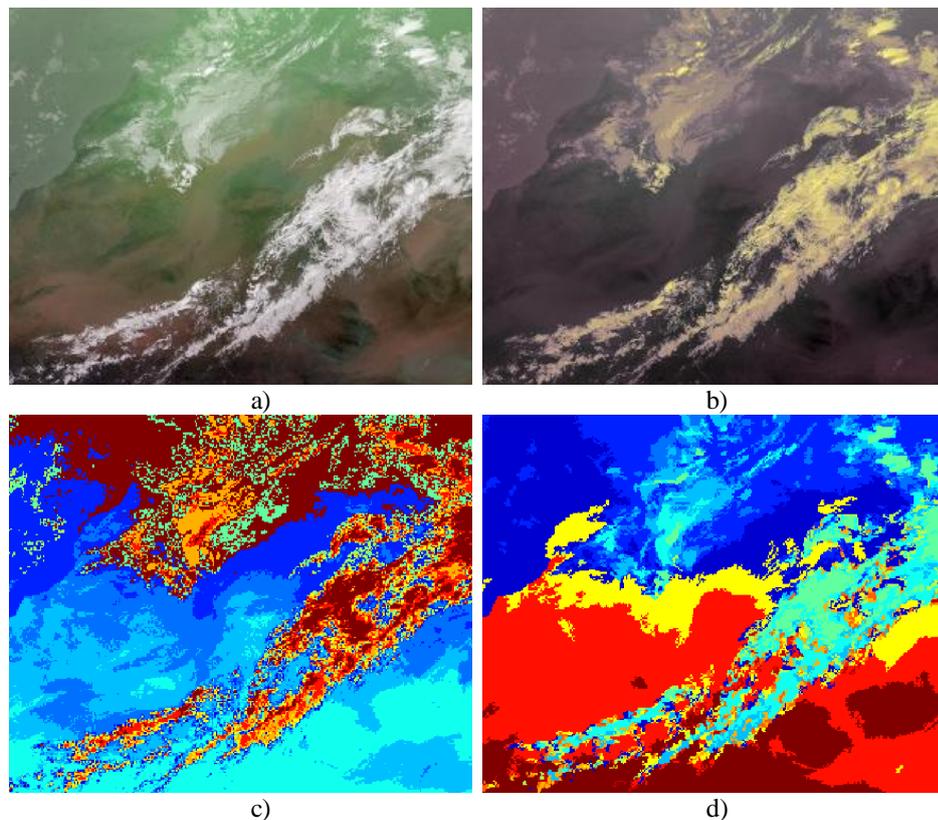


Fig. II.24 : Résultats de la segmentation par les cartes auto-organisatrices de Kohonen de l'image Meteosat prise le 15/12/2006 à 12h00mn en 15 classes  
 a) Composition colorée des composantes IRN8.7, IRN9.7 et IRN10.8 b) Composition colorée des composantes IRN12.0, IRN13.4 et  $(\text{IRN12.0} + \text{IRN13.4})/2$ ; c) résultats de la segmentation vectorielle d) Résultats de la segmentation marginale

## 8 Interprétation et évaluation des résultats

Il existe deux types d'approche pour évaluer les résultats de la segmentation d'image. La première évalue la qualité des résultats de la segmentation en mesurant sa similarité avec une "vérité terrain". Cette "vérité terrain" est souvent subjective, particulièrement dans le cas des images réelles. D'ailleurs, elle exige d'avoir à disposition un expert du domaine d'application. La seconde est basée sur des critères d'évaluation non supervisés, où la qualité des résultats de la segmentation est estimée à partir des statistiques calculées sur chaque zone détectée. Plusieurs de ces critères sont élaborés pour la segmentation d'images en niveau gris et peu d'entre eux sont consacrés pour la segmentation d'images couleur [65, 66] et encore moins pour la segmentation d'images multi-composantes. Les critères élaborés pour les images couleur peuvent être appliqués aux images multi-composantes par simple extension de la dimension des données.

L'un des premiers critères proposés pour les images couleur est la fonction de Liu et de Yang [67]. Borsotti et al. [68] et Zhang et al. [66] ont montré que cette fonction tend à bien évaluer des segmentations très bruitées quand le nombre de régions détectées est important. La

fonction de Liu et Yang a été améliorée en opérant un ajustement sur quelques paramètres [68]. L'inconvénient majeur de cette fonction d'évaluation révisée est qu'elle n'est pas assez sensible aux petites différences de segmentation [66, 68, 69]. Dans ce travail, nous avons choisi d'utiliser la fonction de Borsotti et al. [68] parce qu'elle n'exige aucun réglage de paramètres, elle ne possède pas l'inconvénient de la fonction de Liu et Yang et de leur version révisée et parce qu'elle donne une indication sur la qualité des résultats qui est en adéquation avec notre perception visuelle. Elle est définie par :

$$C_r = \frac{\sqrt{N_R}}{10^4 n} \sum_{r=1}^{N_R} \left( \frac{E_r^2}{1 + \text{Log}(\text{card}B_r)} + \frac{R(B_r)^2}{\text{card}B_r^2} \right) \quad (\text{II.34})$$

avec:

$N_R$ : le nombre de régions dans l'image segmentée,

$n$ : la taille de l'image,

$E_r$ : est la somme des distances entre les pixels de la région  $B_r$  dans l'image d'entrée et les valeurs multi-composantes attribuées à la région  $B_r$  dans l'image segmentée.

$R(B_r)$  : est le nombre de régions ayant exactement une aire égale à celle du  $B_r$ .

Le premier terme de cette somme est élevé seulement pour des régions non homogènes (typiquement, les grandes régions), alors que le second terme est élevé seulement pour les régions  $B_r$  dont l'aire est égale à l'aire de beaucoup d'autres régions dans l'image segmentée (typiquement, les petites régions). Cette fonction pénalise les segmentations qui forment trop de régions et qui ont des régions non homogènes. Plus cette fonction présente une faible valeur, plus la segmentation est de meilleure qualité.

Nous calculons pour chaque résultat obtenu précédemment la valeur de la fonction de Borsotti pour évaluer les segmentations. Les valeurs de cette fonction sont regroupées dans les tableaux 1, 2, 3 et 4. Chacun de ces tableaux regroupe respectivement les résultats obtenus en utilisant les algorithmes EM, K-means, FCM et de Kohonen. Le nombre de classe  $K$  est spécifié à chaque fois sur ces tableaux et nous avons relevé le nombre de régions connexes  $Nr$  obtenu ainsi que le temps de calcul  $Tc$  obtenu sur un PC portable HP Intel Core 2 duo de 1,8 GHz.

Tableau 1 : Résultats des tests en utilisant l'algorithme EM

| image        | Synthétique |       | Avion |       | Fleur |       | Peppers |       | Jelly |       | Meteosat |       |
|--------------|-------------|-------|-------|-------|-------|-------|---------|-------|-------|-------|----------|-------|
| K            | 6           |       | 2     |       | 4     |       | 4       |       | 6     |       | 15       |       |
|              | Vect.       | Marg. | Vect. | Marg. | Vect. | Marg. | Vect.   | Marg. | Vect. | Marg. | Vect.    | Marg. |
| <i>Cr</i>    | 352         | 55    | 221   | 232   | 3646  | 6293  | 1835    | 771   | 621   | 1278  | 7072     | 442   |
| <i>Nr</i>    | 395         | 9     | 4     | 4     | 388   | 23    | 520     | 39    | 919   | 150   | 2753     | 451   |
| <i>Tc(s)</i> | 458         | 1420  | 178   | 579   | 344   | 1037  | 329     | 1036  | 500   | 1964  | 992      | 3641  |

Tableau 2 : Résultats des tests en utilisant l'algorithme K-means

| image        | Synthétique |       | Avion |       | Fleur |       | Peppers |       | Jelly |       | Meteosat |       |
|--------------|-------------|-------|-------|-------|-------|-------|---------|-------|-------|-------|----------|-------|
| K            | 6           |       | 2     |       | 4     |       | 4       |       | 6     |       | 15       |       |
|              | Vect.       | Marg. | Vect. | Marg. | Vect. | Marg. | Vect.   | Marg. | Vect. | Marg. | Vect.    | Marg. |
| <i>Cr</i>    | 157         | 44    | 535   | 223   | 1711  | 9342  | 885     | 659   | 455   | 1304  | 2703     | 260   |
| <i>Nr</i>    | 77          | 6     | 24    | 4     | 385   | 44    | 598     | 80    | 927   | 158   | 5801     | 719   |
| <i>Tc(s)</i> | 0.9         | 74    | 0.3   | 5     | 2     | 65    | 2.86    | 41    | 2     | 103   | 36       | 640   |

Tableau 3 : Résultats des tests en utilisant l'algorithme FCM

| image        | Synthétique |       | Avion |       | Fleur |       | Peppers |       | Jelly |       | Meteosat |       |
|--------------|-------------|-------|-------|-------|-------|-------|---------|-------|-------|-------|----------|-------|
| K            | 6           |       | 2     |       | 4     |       | 4       |       | 6     |       | 15       |       |
|              | Vect.       | Marg. | Vect. | Marg. | Vect. | Marg. | Vect.   | Marg. | Vect. | Marg. | Vect.    | Marg. |
| <i>Cr</i>    | 295         | 94    | 513   | 223   | 1750  | 1079  | 870     | 306   | 443   | 627   | 10656    | 258   |
| <i>Nr</i>    | 274         | 30    | 22    | 4     | 389   | 30    | 592     | 80    | 826   | 110   | 8050     | 818   |
| <i>Tc(s)</i> | 14          | 95    | 6     | 11    | 9     | 25    | 9       | 27    | 11    | 47    | 82       | 359   |

Tableau 4 : Résultats des tests en utilisant l'algorithme SOM

| image        | Synthétique |       | Avion |       | Fleur |       | Peppers |       | Jelly |       | Meteosat |       |
|--------------|-------------|-------|-------|-------|-------|-------|---------|-------|-------|-------|----------|-------|
| K            | 6           |       | 2     |       | 4     |       | 4       |       | 6     |       | 15       |       |
|              | Vect.       | Marg. | Vect. | Marg. | Vect. | Marg. | Vect.   | Marg. | Vect. | Marg. | Vect.    | Marg. |
| <i>Cr</i>    | 517         | 44    | 584   | 216   | 1480  | 935   | 1325    | 1026  | 2236  | 926   | 10311    | 243   |
| <i>Nr</i>    | 779         | 6     | 34    | 4     | 349   | 32    | 466     | 64    | 907   | 164   | 7687     | 891   |
| <i>Tc(s)</i> | 49          | 147   | 47    | 143   | 50    | 172   | 48      | 199   | 49    | 967   | 72       | 885   |

En observant les résultats des segmentations marginales et vectorielles des images couleurs et Meteosat fournis sur les figures II.8, II.9, II.11, II.12, II.14, II.15, II.23 et II.24, nous pouvons constater que la technique marginale développée fournit des résultats assez bon. En effet dans le cas des images couleurs, les résultats sont assez proches de ceux de la segmentation vectorielle ou même améliorés comme le montre l'exemple de l'image "Jelly" en utilisant l'algorithme EM (voir la figure II.8.5.b pour le résultat vectoriel et la figure II.8.5.c pour le résultat marginal).

Visuellement, nous constatons que, la segmentation marginale fournit des résultats moins bons comparativement à la segmentation vectorielle dans les seuls cas de la segmentation de l'image "fleur" par les algorithmes EM et K-means (voir Fig. II.8 et Fig. II.11). Cette constatation est confirmée par les calculs objectifs se trouvant dans les tableaux 1 et 2. En effet, en observant les différents tableaux où une évaluation quantitative des résultats par la fonction de Borsotti  $Cr$  est fournie, nous constatons que 3 résultats sur 6 dans le cas de l'algorithme EM, 4 sur 6 dans le cas de l'algorithme K-means, 5 sur 6 dans le cas de l'algorithme FCM et 6 sur 6 dans le cas de l'algorithme SOM se trouvent améliorés dans le cas de cette segmentation marginale. Soit au total dans 75% des cas, la segmentation marginale proposée donne de meilleurs résultats que la segmentation vectorielle. Pour les images Meteosat, dans tous les cas, le critère de Bosotti se trouve amélioré lors des segmentations marginales car la technique de fusion utilisée commence par éliminer les régions de très faible dimensions et c'est effectivement ces régions, comme le stipule Borsotti et al. [68], qui pénalisent leur fonction. En observant toujours les différents tableaux, nous constatons aussi que le nombre de régions se trouve au mieux réduit, au plus identique lors de la segmentation marginale par rapport à la segmentation vectorielle, bien sur cela est dû à la technique de fusion.

Les temps de calcul dans le cas de la segmentation marginale se trouvent augmentés par rapport à la segmentation vectorielle. Ce résultat est prévisible car l'application de l'algorithme de segmentation dans le cas marginal est réalisé autant de fois que l'image compte de composantes, 3 pour l'image couleur et 5 pour l'image Meteosat, contre une seule fois en segmentation vectorielle, ajouter à cela le temps nécessaire pour la fusion.

## 9 Conclusion

Dans ce chapitre, nous avons développé deux stratégies de segmentation des images multi-composantes, à savoir la stratégie marginale et la stratégie vectorielle et nous les avons appliquées à l'imagerie couleur et satellitaire. Dans la stratégie marginale, nous avons

développé une nouvelle technique de fusion des cartes des étiquettes. Pour rendre compte de l'efficacité de cette technique, plusieurs algorithmes de segmentation telle que EM, K-means, FCM, et SOM ont été adaptés. Les résultats des tests sont comparés à ceux obtenus par les mêmes algorithmes dans le cas d'une stratégie de segmentation vectorielle. Le critère choisi pour cette comparaison est la fonction de Borsotti. Cette évaluation quantitative, réalisée sur plusieurs images, montre que la technique marginale que nous avons développée fournit des résultats appréciables.

## Chapitre III

# Approche markovienne utilisant plusieurs formes d'énergies de Gibbs pour la segmentation des images de télédétection

### 1 Introduction

Le développement de nouveaux capteurs multi-spectraux en imagerie de télédétection permet l'acquisition d'images multidimensionnelles d'une grande richesse, néanmoins, leurs interprétations se heurtent au problème de traitement du grand nombre de données multidimensionnelles.

La segmentation des images est largement employée en télédétection particulièrement depuis la disponibilité des images de très haute résolution. De nombreuses approches de segmentation d'images de télédétection basées sur les algorithmes des colonies de fourmi [70], mean-shift [71, 72], ligne de partage des eaux [73, 74], classification floue [75], memetic [74], fractale et MRF [76-80] ont été développées.

Rappelons que les trois approches nommées marginale, scalaire et vectorielle sont applicables à ces images multi-spectrales. Dans l'approche vectorielle, la segmentation est directement réalisée sur l'image multi-spectrale en utilisant une information vectorielle pour chaque pixel. Bien que cette dernière soit plus complexe à mettre en œuvre, notre choix s'est porté sur elle car elle présente l'avantage majeur de prendre en considération les interactions entre les différentes composantes spectrales de l'image.

Nous traitons dans ce chapitre des images réelles donc naturelles et la nature n'est pas entièrement "imprévisible", deux observations très éloignées l'une de l'autre ont en moyenne pas d'influence entre elles, alors que deux observations très voisines s'influencent plus largement qu'elles ne le sont plus proches [79, 81]. Cette notion est assez proche de la propriété de Markov qui stipule que la loi conditionnelle de la valeur d'un site dépend seulement de ses sites voisins. Ainsi, la probabilité qu'un pixel appartienne à une classe dépend non seulement de son intensité, mais aussi de celles de ses voisins [82, 83].

Nous utilisons donc dans ce travail, une approche vectorielle de la segmentation par les champs de Markov prenant ainsi en compte les interactions spectrales et spatiales entre les pixels des images de télédétection du satellite Meteosat Second Génération 2 (MSG2).

Quelques travaux traitent de la segmentation des images de télédétection par les champs aléatoires de Markov.

Par exemple, B. Tso et R.C. Olsen [78] emploient les champs de Markov pour segmenter les images du satellite IKONOS afin de réaliser une cartographie du parc naturel Elkhorn situé sur la côte californienne. L'optimisation est réalisée par la maximisation de la probabilité à posteriori (MAP) et l'initialisation est obtenue par une pré-classification en ondelettes.

J. Keuche et al. [76] réalisent aussi une segmentation markovienne des images de Landsat 5 dans un but de cartographier la région Tenerife des îles Canarie. L'optimisation de la probabilité à posteriori est effectuée par une approche de type déterministe par l'algorithme ICM (*Iterated Conditional Modes*). La classification markovienne fait suite à une pré-segmentation par deux méthodes de classification supervisée utilisant des données recueillies au sol. Les résultats dans ce cas se trouvent améliorés pour un faible nombre d'itérations, mais dégradés au fur et à mesure que le nombre d'itérations devient important car des petites régions sont négligées.

Habituellement, en utilisant la segmentation markovienne, le processus image est modélisé par une énergie gaussienne, et le processus étiquette est modélisé par une énergie de Gibbs. L'énergie de Gibbs qui est souvent employée est celle dite de Pappas. Dans ce chapitre, nous proposons de comparer les plus connues des énergies de Gibbs sur la segmentation vectorielle d'images multi-spectrales MSG2. Nous proposons aussi deux nouvelles énergies que nous évaluerons en même temps que les autres [79, 84].

Nous traitons dans ce chapitre de la segmentation d'images par les champs de Markov et comme les champs markovien sont, de part leur définition, intrinsèquement liées à la notion de système de voisinage qui constitue une notion de base en imagerie, l'introduction de quelques notions élémentaires utiles pour une meilleure compréhension de l'approche adoptée nous paraît nécessaire. Nous définirons ainsi les notions d'image, de voisinage et de clique avant de développer la notion de champs de Markov et de Gibbs. Nous appliquerons par la suite la théorie des champs Markoviens à la segmentation des images MSG2. Les résultats expérimentaux seront analysés et évalués en utilisant la fonction de Borsotti [68].

## **2 Notions d'image, de voisinage et de clique**

Les champs que nous utilisons dans ce paragraphe ont été introduits par des physiciens et pour cette raison, nous altérons légèrement notre langage et appelons un site ce que nous avons l'habitude d'appeler, en traitement d'image, un pixel.

## 2.1 Notion d'image

L'image est représentée par une grille rectangulaire finie bidimensionnelle de taille  $L \times M$  ( $L$  étant le nombre de lignes et  $M$  le nombre de colonnes), dont l'ensemble  $S$  de sites  $s$  peuvent être désignés soit par leurs coordonnées  $S = \{s(i,j) ; 1 \leq i \leq L, 1 \leq j \leq M\}$  soit par leur numéro d'ordre  $S = \{s_1, s_2, \dots, s_n\}$ , tel que  $n = L \times M$  est le cardinal de  $S$ .

A chaque site  $s$  est associée une variable aléatoire  $y_s$ , dont les valeurs  $y_s$  appartiennent à un ensemble  $\Psi$ . Par exemple  $\Psi = \{0, 1, 2, \dots, 255\}$  représentant les valeurs des niveaux de gris dans le cas d'images monochromes et  $\Psi = \{0, 1, 2, \dots, 255\}^D$  correspondant aux valeurs de dimension  $D$  associés à chaque site dans le cas d'images multi-composantes à  $D$  composantes.

L'image est alors considérée comme un vecteur aléatoire  $y = (y_s, s \in S)$  dont le vecteur  $y = (y_s, s \in S)$  est une configuration (ou réalisation). Une configuration est un élément de l'espace produit  $\Gamma = \Psi^{L \times M}$ .

## 2.2 Notion de Voisinage

On se donne un ensemble des sites  $S$ , et on définit une relation de voisinage  $V = \{V_s, s \in S\}$  en terme de "plus proches voisins" sur cette ensemble  $S$  dont les propriétés sont :

$$\begin{aligned} \forall s \in S, s \notin V_s \quad \text{et} \\ \forall \{s, t\} \subset S, s \in V_t \Leftrightarrow t \in V_s. \end{aligned} \quad (\text{III.1})$$

L'ensemble  $V_s$  est appelé le voisinage de  $s$ .  $t$  est dit voisin de  $s$  si  $t \in V_s$ . Cette relation est symétrique. Ainsi, si on considère l'ensemble des voisins d'un pixel  $s$  d'une image comme l'ensemble des sites  $t$  dont la distance entre  $s$  et  $t$  est inférieure à une constante fixe  $\varepsilon$ , le système de voisinage de  $s$  est donné par :

$$V_s = \{t \in S / d(t, s) \leq \varepsilon\} \quad (\text{III.2})$$

Les systèmes de voisinages les plus courants sont ceux d'ordre 1 et 2 définies relativement aux distance euclidiennes 1 et  $\sqrt{2}$ . La figure III.1 représente les systèmes de voisinage du 1<sup>er</sup> ordre avec les quatre plus proches voisins et du 2<sup>ème</sup> ordre avec les huit plus proches voisins.

## 2.3 Notion de cliques

Une fois fixée la relation de voisinage  $V$ , on construit l'ensemble  $C$  des cliques  $c$  de  $S$  relativement à ce système de voisinage. En effet, on appelle clique tout sous-ensemble  $c$  qui peut être un site unique ou un ensemble de sites qui sont mutuellement voisins. Par exemple,

si  $s$  et  $t$  forment une clique ( $\{s, t\} \subset C$ ) alors  $t \in V_s$  et  $s \in V_t$ . L'ordre d'une clique correspond au nombre de ses éléments. Le nombre et les ordres des cliques dépendent du système de voisinage utilisé. Pour le système de voisinage de 1<sup>er</sup> ordre avec 4 types de connectivités, nous avons des cliques d'ordre 1 et d'ordre 2. Pour un système de voisinage de 2<sup>ème</sup> ordre avec 8 connectivités nous avons des cliques d'ordre 1, 2, 3 et 4 (voir Fig. III.1).

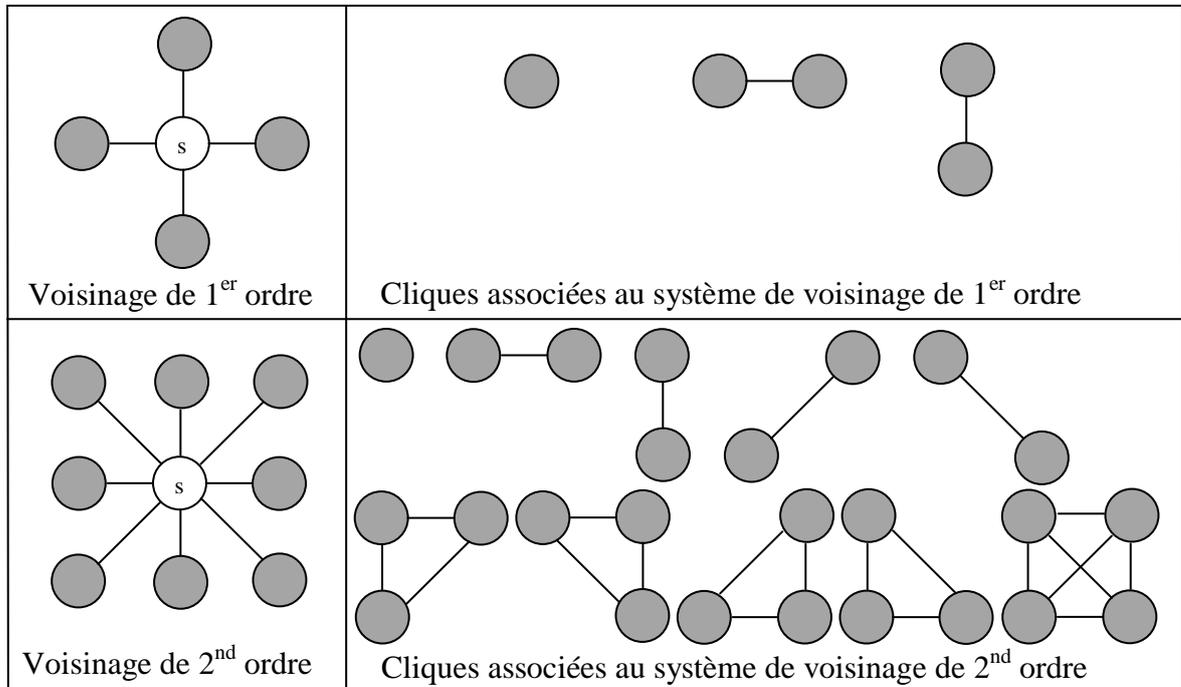


Fig. III.1 : Voisinage bidimensionnel d'ordres 1 et 2 et les cliques associées

### 3 Chaîne et champs de Markov

La théorie des champs de Markov est une branche de la théorie des probabilités permettant l'analyse des dépendances spatiales de phénomènes physiques.

Soit une suite de variables aléatoires  $X = \{x_s, s = 1, 2, \dots\}$  définie dans le temps, cette chaîne est dite de Markov si elle vérifie la propriété de Markov. Cette propriété dit que la loi conditionnelle de la valeur d'aujourd'hui, étant donnée toutes les valeurs du passé, ne dépend que de la valeur d'hier (voir Fig. III.2). Cette propriété s'écrit :

$$P(x_s | x_{s-1}, x_{s-2}, \dots, x_1) = P(x_s | x_{s-1}) \quad (\text{III.3})$$

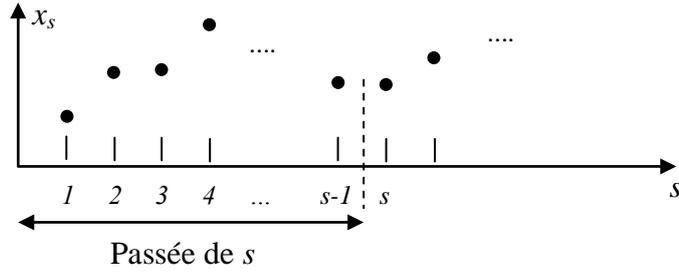


Fig. III.2 : Une chaîne de Markov

On peut aussi donner une version symétrique de cette propriété qui dit que la loi conditionnelle de la valeur en un instant  $s$ , étant donné toutes les autres valeurs, ne dépend que des valeurs des deux instants voisins  $s-1$  et  $s+1$ .

$$P(x_s | x_t, t \neq s) = P(x_s | x_{s-1}, x_{s+1}) \quad (\text{III.4})$$

En généralisant la propriété de Markov à dimension 2, on obtient un champ de Markov. Ainsi, dans le cas d'une image, le monde extérieur à un site  $s$  n'agit sur lui que par ses plus proches voisins relativement à un système de voisinage.

Donc un champ aléatoire  $\mathcal{Y} = \{Y_s, s \in S\}$  est un champ markovien sur  $S$  relativement à  $V$  si il vérifie la propriété suivante :

$$\forall s \in S, P(Y_s = y_s / Y_t = y_t, t \in S \text{ et } t \neq s) = P(Y_s = y_s / Y_t = y_t, t \in V_s) \quad (\text{III.5})$$

Cela signifie que la connaissance d'un voisinage local  $V_s$  du pixel  $s$  est suffisante pour calculer la probabilité du pixel  $s$ . Dans la pratique, l'introduction des champs markoviens permet la modélisation d'ensembles de variables aléatoires dont les interactions mutuelles résultent uniquement de la combinaison d'interactions locales. Il est clair que l'intérêt porté aux champs de Markov réside essentiellement dans la notion de localité (un voisinage de faible taille).

#### 4 Champs de Gibbs et Théorème de Hammersley-Clifford

La distribution de Gibbs est une représentation d'une mesure  $\pi$  positive sur  $\Gamma$ . La distribution statistique de Gibbs est définie par sa fonction d'énergie  $E$  donnée par un ensemble de fonctions potentielles  $v_c$ . Les éléments essentiels d'une fonction potentielle sont :

$$\begin{cases} v_\emptyset(y) = 0 \\ v_c(y) = v_c(y') \text{ si } y_s = y'_s \forall s \in S \end{cases} \quad (\text{III.6})$$

Ainsi, un champ aléatoire  $\mathcal{Y}=\{y_s, s \in S\}$  est un champ de Gibbs sur  $S$  relativement à un système de voisinage  $V$  si et seulement si sa distribution de probabilité  $P(\mathcal{Y}=y)$  est une distribution de Gibbs  $\pi$  donnée par :

$$\forall y \in \Gamma, P(\mathcal{Y}=y)=\pi(\mathcal{Y}=y)=\frac{1}{Z} \exp(-E(y)) \quad (\text{III.7})$$

avec :  $E(y)$  est la fonction d'énergie du Champs de Gibbs  $\mathcal{Y}$

$Z$  est une constante de normalisation appelée aussi fonction de partition.

La constante de normalisation est donnée par la fonction :

$$Z = \sum_{y \in \Gamma} \exp(-E(y)) \quad (\text{III.8})$$

La fonction d'énergie est donnée par la relation :

$$E(y) = \sum_{c \in C} v_c(y) \quad (\text{III.9})$$

avec  $v_c$  qui sont des fonctions potentiel qui ne dépendent que des éléments de  $y$  qui appartiennent à la clique  $c$ .

Le champ est dit homogène si les fonctions de potentiels sont indépendantes de la position des cliques, et il est dit isotrope si elles sont indépendantes de leur orientation.

Une propriété fondamentale d'un champ de Gibbs est que les interactions entre sites sont locales, c'est-à-dire limitées à celles entre les voisins. De façon heuristique, cela exprime le fait que si une perturbation se produit, elle a peu d'impact sur les sites éloignés de la perturbation [85] (voir Fig. III.3).

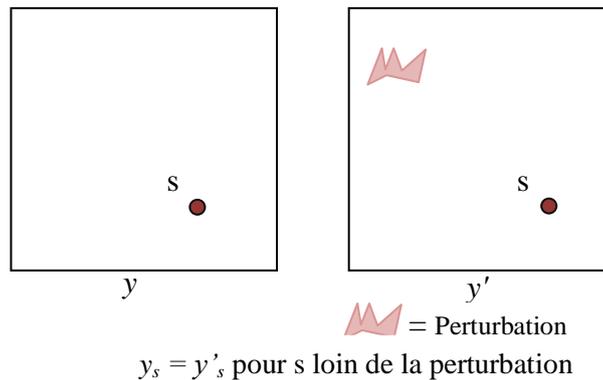


Fig. III.3 : Illustration d'une interaction locale

En effet, pour formuler cette propriété, déterminons la loi conditionnelle au site  $s$ , étant donnée son extérieur. On aura ainsi :

$$P_s(y_s / y_t, \forall t \neq s) \equiv \exp \left( - \sum_{\substack{c \in C, \\ c \text{ clique de } V_s}} v_c(y_s) \right) \quad (\text{III.10})$$

On voit bien que seules les cliques du voisinage  $V_s$  contribuent à cette loi conditionnelle.

Un champ de Gibbs est caractérisé par sa propriété globale (distribution de Gibbs) qui s'exprime par le fait que si une perturbation se produit, elle a peu d'impact sur les sites éloignés de la perturbation. Un champ de Markov est caractérisé par sa propriété markovienne qui stipule que le monde extérieur à un site  $s$  n'agit sur lui que par ses plus proches voisins relativement à un système de voisinage.

L'équivalence entre champs de Gibbs et champs de Markov a été établie en premier lieu par Hammersley-Clifford [86] sans que cela ne soit publié. Besag en 1974 dans son article [82] leur attribue cette équivalence. Depuis, le théorème de Hammersley-Clifford est toujours référencé à l'article de Besag. Le théorème de Hammersley-Clifford [82] s'énonce comme suit : Un champ aléatoire sur l'ensemble fini de sites  $S$  est un champ de Markov relativement au système de voisinage  $V$  vérifiant la propriété de positivité si et seulement si c'est un champ de Gibbs de potentiels associé à  $V$ .

## 5 Segmentation markovienne d'images multi-composantes

### 5.1 Formulation du problème de la segmentation d'images

La segmentation d'une image par classification revient à répartir l'ensemble des pixels  $S$  en classes homogènes. La classification est considérée donc comme l'estimation d'un processus étiquette (label)  $\mathcal{X}$  à partir de la réalisation d'un processus pixel  $\mathcal{Y}$ . Soit  $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \dots\}$  le champ des classes. Pour chaque site  $s$  de  $S$ ,  $x_s$  prend sa valeur dans l'ensemble des classes  $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$  avec  $K$  est le nombre des classes. L'image des classes n'est pas directement accessible. Segmenter une image, revient alors à attribuer une estimation  $\hat{x}$  du champ  $\mathcal{X}$  inobservable ou "caché" à partir de l'observation  $y$  du champ aléatoire observé  $\mathcal{Y}$ .

Dans ce cas, la modélisation probabiliste peut être d'une remarquable efficacité. On peut utiliser une approche Bayésienne qui consiste à rechercher la classe la plus probable au sens de la probabilité à posteriori obtenue par la maximisation de  $P(\mathcal{X}=x \mid \mathcal{Y}=y)$ . Il s'agit donc de construire  $P(\mathcal{Y}=y \mid \mathcal{X}=x)$ , puis fixer une loi a priori sur  $\mathcal{X}$ , notée  $P(\mathcal{X}=x)$ , à partir de laquelle on calcule la probabilité a posteriori  $P(\mathcal{X}=x \mid \mathcal{Y}=y)$  par la formule de Bayes suivante [85, 87- 89]:

$$P(\mathcal{X} = x \setminus \mathcal{Y} = y) = \frac{P(\mathcal{Y} = y \setminus \mathcal{X} = x)P(\mathcal{X} = x)}{P(\mathcal{Y} = y)} \quad (\text{III.11})$$

où :  $P(\mathcal{X} = x \setminus \mathcal{Y} = y)$  est la loi a posteriori,

$P(\mathcal{Y} = y \setminus \mathcal{X} = x)$  représente la probabilité conditionnelle de l'observation  $y$  sachant la classification  $x$ ,

$P(\mathcal{X} = x)$  représente la loi de probabilité associée au champ  $\mathcal{X}$ , encore appelée loi a priori,

$P(\mathcal{Y} = y)$  est constante car  $c$ 'est une réalisation du processus observé.

On se passe de  $P(\mathcal{Y} = y)$  car pour  $y$  fixé, on cherche  $x$ , ainsi ce terme n'intervient pas dans la maximisation de la loi à posteriori. Maximiser  $P(\mathcal{X} = x \setminus \mathcal{Y} = y)$  revient donc à maximiser  $P(\mathcal{Y} = y \setminus \mathcal{X} = x) P(\mathcal{X} = x)$

$$\begin{aligned} \hat{x} &= \text{Arg}(\max_{x \in \Omega} (P(\mathcal{X} = x \setminus \mathcal{Y} = y))) \\ &= \text{Arg}(\max_{x \in \Omega} (P(\mathcal{Y} = y \setminus \mathcal{X} = x)P(\mathcal{X} = x))) \end{aligned} \quad (\text{III.12})$$

Pour obtenir la classification la plus probable  $\hat{x}$  en utilisant la maximisation de la probabilité a posteriori (MAP), il est nécessaire de réaliser les deux étapes suivantes :

- La première étape est la modélisation du champ des observations et celle du champ des classes, c'est-à-dire trouver le modèle statistique de  $P(\mathcal{Y} = y \setminus \mathcal{X} = x)$  et le modèle de  $P(\mathcal{X} = x)$ .
- La deuxième étape est l'optimisation de la probabilité a posteriori.

## 5.2 Modélisation du champ des observations

En absence d'informations sur la nature de la distribution des observations, il est d'usage d'utiliser une loi gaussienne afin de modéliser la densité de probabilité conditionnelle  $P(\mathcal{Y} = y \setminus \mathcal{X} = x)$  [75, 78, 83-85, 88].

Dans la cas d'une image contenant une seule composante, la probabilité  $P(\mathcal{Y}_s = y_s \setminus \mathcal{X}_s = k)$  (avec  $y_s$  qui désigne la valeur observée du site  $s$  et  $k$  la valeur prise par l'étiquette de ce site) peut être modélisée sous la forme suivante :

$$P(\mathcal{Y}_s = y_s \setminus \mathcal{X}_s = k) = \frac{1}{\sqrt{2\pi \Sigma_k^2}} \exp\left(-\frac{(y_s - \bar{y}_k)^2}{2 \Sigma_k^2}\right) \quad (\text{III.12})$$

où  $\Sigma_k^2$  est la variance de classe  $k$  et  $\bar{y}_k$  est sa moyenne. La détermination de cette probabilité est liée à l'ensemble des moyennes et des variances des classes.

Dans la cas d'une image contenant  $D$  composantes, la probabilité  $P(\mathcal{Y}_s=y_s/\mathcal{X}_s=k)$  est modélisée par une loi gaussienne multidimensionnelle donnée par la relation :

$$P(\mathcal{Y}_s = y_s \setminus \mathcal{X}_s = k) = \frac{1}{(2\pi|\Sigma_k|)^{D/2}} \exp\left(-\frac{1}{2}(y_s - \bar{y}_k)^T \Sigma_k^{-1} (y_s - \bar{y}_k)\right) \quad (\text{III.13})$$

où :

$y_s$  est le vecteur constitué par les valeurs des  $D$  différentes composantes du site  $s$ .

$\Sigma_k$  est la matrice des variances covariances entre les différentes composantes pour la classe  $k$ ,  $\bar{y}_k$  est le vecteur moyennes pour la classe  $k$ .

$D$  est le nombre de composantes présentes dans l'image à segmenter.

La détermination de cette probabilité est liée à l'ensemble des vecteurs de moyennes et des matrices de variances covariances des classes.

### 5.3 Modélisation du champ des classes

Le champ de classes  $x$  est considéré comme un champ aléatoire de Markov par rapport au système de voisinage choisie, alors :

$$P(\mathcal{X}_s = x_s \setminus \mathcal{X}_t = x_t, \forall t \in \mathcal{S}, t \neq s) = P(\mathcal{X}_s = x_s \setminus \mathcal{X}_t = x_t, \forall t \in \mathcal{V}_s) \quad (\text{III.14})$$

Le théorème de Hammersley-Clifford [82] ayant établie une équivalence entre champ de Markov et champ de Gibbs, le champ des classes est aussi un champ de Gibbs dont la distribution est donnée par :

$$\pi(\mathcal{X}_s = x_s) = \frac{1}{Z} \exp(-E(x_s)) = \frac{1}{\sum_{x \in \Omega} \exp(-E(x))} \exp\left(-\sum_{c \in \mathcal{C}} v_c(x_s)\right) \quad (\text{III.15})$$

où  $E(x_s)$  est la fonction d'énergie du champ de Gibbs  $\mathcal{X}$  qui est obtenue sous la forme d'une somme de potentiels locaux  $v_c(x_s)$  sur les cliques  $c$  de  $\mathcal{C}$ . La fonction de potentiels permet de caractériser la différence ou la ressemblance entre les pixels de chaque clique.

Le choix des fonctions d'énergie est un problème difficile à résoudre pour obtenir de bons résultats avec un algorithme de classification donné. Il faut que la formule définissant l'énergie permette des classifications correctes sans entraîner des coûts importants en temps de calcul. La littérature présente plusieurs modèles de ces fonctions [79, 83, 90-93]. Nous allons dans ce travail utiliser cinq d'entre elles. Nous proposons aussi deux nouveaux

modèles. Les résultats obtenus seront par la suite comparés. Dans tous ces modèles, la fonction d'énergie est calculée à partir de potentiels issus seulement des cliques d'ordre deux d'un système de voisinage correspondant au 8 plus proches voisins.

La première fonction que nous utiliserons est celle de Pappas [93]. Elle est aussi connue sous le nom de modèle de Potts dans les statistiques employées en physique. Elle est donnée par la relation :

$$E1(x_s) = \sum_{c \in \mathcal{C}} v_c(x_s) = \sum_i \sum_j \beta_{ij} \quad (\text{III.16})$$

avec  $\{i,j\} \subset \mathcal{C}$  c'est-à-dire  $\{i,j\}$  sont les paires de pixels formant des cliques et correspondant au voisinage de  $s$ . Le potentiel de clique  $\beta_{ij}$  associé aux différentes paires de pixels est donné par :

$$\beta_{ij} = \begin{cases} \beta & \text{si } x_i \neq x_j \\ -\beta & \text{si } x_i = x_j \end{cases} \quad (\text{III.17})$$

où  $\beta$  est un paramètre positif. Le rôle de ce paramètre est de favoriser l'homogénéité spatiale entre pixels voisins. Pappas choisit une valeur de  $\beta$  égale à 0.5 [93]. C'est cette valeur qui sera retenue dans ce travail.

Par soucis de généralisation, plusieurs fonctions sont regroupées dans une même écriture qui est donnée sous la forme suivante :

$$E2(x_s) = \sum_{c \in \mathcal{C}} v_c(x_s) = \beta \sum_i \sum_j |x_i - x_j|^p \quad (\text{III.18})$$

avec  $\beta$  qui est un paramètre de régularisation de lissage et  $p$  la puissance utilisée dans la fonction.

Les valeurs de  $p=1$  et  $p=2$  correspondent respectivement à la fonction de la valeur absolue de Lamotte [91] et à la fonction quadratique utilisée par Blanc-Féraud [92]. Nous proposons dans cette étude d'ajouter la fonction correspondant à une valeur de  $p=1/2$ . Celle-ci constitue l'une des nouvelles fonctions que nous proposons [79].

Une autre fonction utilisée par Lamotte [91] est donnée par :

$$E3(x_s) = \sum_{c \in \mathcal{C}} v_c(x_s) = \beta \sqrt{\frac{\sum_i \sum_j |x_i - x_j|}{cst}} \quad (\text{III.19})$$

où  $cst$  est une constante de normalisation.

La fonction proposée par Geman and Geman [83] est la suivante :

$$E4(x_s) = \sum_{c \in \mathcal{C}} v_c(x_s) = \beta \sum_i \sum_j \frac{(x_i - x_j)^2}{1 + (x_i - x_j)} \quad (\text{III.20})$$

Dans notre application, sachant que pour modéliser les précipitations les fonctions sphériques sont souvent utilisées [94, 95] et sachant que les précipitations sont étroitement liées aux nuages, nous proposons de tester une nouvelle fonction d'énergie de forme sphérique [79, 90] que nous donnons par la relation :

$$E5(x_s) = \sum_{c \in \mathcal{C}} v_c(x_s) = \beta \sum_i \sum_j \left| 1.5(x_i - x_j) - 0.5(x_i - x_j)^3 \right| \quad (\text{III.21})$$

Nous attribuons la valeur  $I$  au paramètre de régularisation de lissage pour les fonctions de potentiels  $E2$ ,  $E3$ ,  $E4$ , et  $E5$ . Toutes les fonctions de potentiels citées précédemment imposent une contrainte de connectivité spatiale sur la classification. L'affectation du potentiel permet de donner une probabilité élevée pour les paires de pixels ayant des étiquettes identiques et une probabilité faible pour les paires de pixels ayant des étiquettes distinctes

#### 5.4 Fonction d'énergie globale

Rappelons que maximiser  $P(\mathcal{X}=x | \mathcal{Y}=y)$  revient à maximiser le produit  $P(\mathcal{Y}=y | \mathcal{X}=x)P(\mathcal{X}=x)$  (eq. III.12). Comme ces deux dernières probabilités sont des fonctions exponentielles et pour faciliter les calculs, nous posons :

$$\begin{cases} U_1(y|x) = -\ln P(\mathcal{Y}=y | \mathcal{X}=x) \\ U_2(x) = -\ln P(\mathcal{X}=x) \end{cases} \quad (\text{III.22})$$

L'estimateur du maximum a posteriori devient alors :

$$\begin{aligned} \hat{x} &= \text{Arg} \left[ \max_{x \in \mathcal{X}} \left[ \exp \left( -(U_1(y|x) + U_2(x)) \right) \right] \right] \\ &= \text{Arg} \left[ \min_{x \in \mathcal{X}} (U_1(y|x) + U_2(x)) \right] \\ &= \text{Arg} \left[ \min_{x \in \mathcal{X}} (U_G(x, y)) \right] \end{aligned} \quad (\text{III.23})$$

Donc, la configuration de la classe la plus probable est celle qui présente une énergie globale  $U_G$  minimale, avec, dans le cas d'images multi-composantes :

$$U_G(k, y_s) \approx \frac{1}{2} (y_s - \bar{y}_k)^T \Sigma_k^{-1} (y_s - \bar{y}_k) + \frac{D}{2} \ln(2\pi |\Sigma_k|) + \sum_i \sum_j \beta_{ij} \quad (\text{III.24})$$

### 5.5 Optimisation de la probabilité a posteriori

La classification au sens de la maximisation a posteriori (MAP) consiste à déterminer la valeur de la classe la plus probable  $\hat{x}$  qui maximise  $P(\mathcal{X}|\mathcal{Y})$  ou de manière équivalente, qui minimise la fonction d'énergie globale  $U_G(x,y)$ . Dans la littérature, cette optimisation peut être effectuée soit par des méthodes stochastiques, soit par des méthodes déterministes. Les méthodes stochastiques sont très longues du point de vue du temps de calcul. Les méthodes itératives déterministes, bien que ne fournissant pas toujours l'optimum global, sont fréquemment utilisées que les méthodes stochastiques parce qu'elles sont plus rapides.

Parmi les algorithmes déterministes, nous avons choisi l'algorithme ICM (*Iterated Conditional Modes*) [87]. Celui-ci consiste à chercher le minimum de la fonction d'énergie en suivant les étapes données en figure III.4.

1. Initialisation : il s'agit de fournir à l'algorithme une image classifiée et d'estimer les paramètres de la classification initiale. Les paramètres correspondent à la moyenne et à la matrice variances covariances de chaque classe.
2. A partir d'une configuration initiale, les étapes suivantes sont réalisées pour chaque pixel:
  - Pour chaque classe, calculer l'énergie globale  $U_G$  (voir équation III.24)
  - Rechercher la classe donnant une énergie globale minimale.
  - Affecter l'étiquette de cette classe au site courant.
3. On vérifie si le critère d'arrêt est satisfait. Si c'est le cas, l'algorithme s'arrête sinon on retourne à l'étape 2.

Fig.III.4 : Algorithme ICM

L'algorithme ICM est un algorithme itératif et permet de minimiser la fonction d'énergie de chaque pixel, connaissant la valeur observée du pixel et la classe courante de tous les autres pixels à l'intérieur de la fenêtre du voisinage. L'ensemble des pixels de la fenêtre du voisinage est utilisé dans le calcul de la probabilité a priori basée sur la loi de Gibbs.

La maximisation de l'algorithme est effectuée pour chaque point de l'image et le cycle est répété jusqu'à satisfaction du critère d'arrêt. Le critère d'arrêt peut être, soit un nombre d'itérations fixé par l'utilisateur, soit un seuil à ne pas dépasser sur le nombre de pixels ayant changé de classe à la dernière itération. Généralement, ce seuil est de l'ordre de 10% du nombre de pixels dans la scène [83, 92, 93]. L'algorithme ICM possède l'avantage de converger rapidement.

## 6 Les images tests

Les images utilisées dans ce chapitre sont fournies par le satellite météorologique Meteosat seconde génération 2 (MSG 2) (voir chapitre I). Son instrument principal est un radiomètre imageur, SEVIRI, équipé de 12 canaux spectraux. Celui-ci permet de mesurer l'énergie transportée par les ondes électromagnétiques à des fréquences spectrales données. Un grand nombre de paramètres météorologiques (comme les précipitations, la direction et la vitesse des vents, ...) peut être estimé par l'étude des densités des nuages, leur température de brillance, leur mouvement, etc. [94-97]. Il faut alors au préalable extraire et caractériser les nuages présents dans ces images en utilisant les canaux appropriés.

Les canaux visibles VIS0.6 (0.56 $\mu$ m-0.71 $\mu$ m), VIS0.8 (0.74 $\mu$ m-0.88 $\mu$ m), le canal visible haute résolution HRV (0.5 $\mu$ m-0.9 $\mu$ m) et les canaux proches infrarouges NIR1.6 (1.5 $\mu$ m-1.78 $\mu$ m) et NIR3.8 (3.48 $\mu$ m-4.36 $\mu$ m) sont bien adaptés à l'étude des nuages bas ou des nuages hauts et épais. Cependant, il faut noter que ces canaux dépendent intégralement de la lumière incidente, donc ils ne sont exploitables qu'en journée. De plus, vu que les surfaces couvertes sont étendues, il est nécessaire d'effectuer une correction en fonction de la position du soleil afin d'avoir un éclairage homogène sur toute la scène. Pour ces différentes raisons, ces canaux seront écartés de cette étude.

Les canaux vapeur d'eau VW6.2 (5.35 $\mu$ m-7.15 $\mu$ m) et VW7.3 (6.85 $\mu$ m-7.85 $\mu$ m) n'ont pas été retenus aussi car ils ne caractérisent pas les nuages convectifs qui sont généralement les nuages précipitants.

Les cinq canaux du domaine infrarouge IR8.7 (8.3 $\mu$ m-9.1 $\mu$ m), IR9.7 (9.38 $\mu$ m-9.94 $\mu$ m), IR10.8 (9.8 $\mu$ m-11.8 $\mu$ m), IR12.0 (11 $\mu$ m-13 $\mu$ m), et IR13.4 (12.4 $\mu$ m-14.4 $\mu$ m) ont été retenus. Ils sont effectivement très intéressants dans notre étude car contrairement aux canaux visibles, ils sont exploitables de jour comme de nuit et parce qu'ils donnent la température de brillance des nuages épais [97]. En effet les nuages convectifs dont les sommets se situent très haut dans l'atmosphère fournissent la majorité des précipitations car ils sont les plus froids. Les cirrus, autres nuages de très haute altitude, peuvent être différenciés des amas convectifs parce que la température qu'ils affichent au travers de ces capteurs est plus chaude. A cause de leur faible épaisseur, leur température de brillance est combinée avec celles de couches atmosphériques moins élevées en altitude. Au niveau de leurs frontières, les amas convectifs ont donc, sur les composantes images infrarouges thermiques, un gradient très élevé. Ainsi, on peut espérer une bonne détection de la forme des amas convectifs en utilisant ces canaux [94, 96, 97].

On se propose d'étudier la région nord ouest de l'Afrique qui est comprise entre les latitudes

15° nord à 40° nord et les longitudes 14° ouest à 16° Est. Elle englobe la totalité des territoires algérien, tunisien, marocain, saharien occidental et mauritanien, et une partie du Mali, du Niger et de Libye. Dans ces images, nous pouvons aussi distinguer une partie de la Méditerranée de l'océan atlantique et du sud de l'Europe (voir Fig. III.5).



Fig. III.5 : Image du canal VIS08 prise le 01/09/2006 à 12H00

Les images étudiées correspondent à la période hivernale de cette région où une forte activité atmosphérique peut être constatée. L'apparition, le mouvement et la disparition des nuages sur le nord ouest de l'Afrique peuvent être ainsi observés. Notons que chaque canal de ces images d'étude MSG2 constitue une composante spectrale qui peut être considérée comme une image en niveau de gris de dimension  $600 \times 500$  pixels<sup>2</sup> et qui est codée sur 8 bits.

## 7 Résultats expérimentaux

La segmentation markovienne décrite par l'algorithme ICM nécessite des valeurs initiales des vecteurs des moyennes et des matrices de covariances de chaque classe. La classification initiale de l'image d'entrée est réalisée, en major partie, par l'algorithme EM [75, 98]. Cet algorithme est très sensible aux conditions initiales. Comme le but de ce travail est de comparer les résultats de la segmentation en utilisant différentes formes d'énergie de Gibbs, ces résultats doivent être indépendants de l'étape d'initialisation. Par conséquent, nous proposons une classification initiale simple rapide pour l'algorithme ICM. La classification initiale est calculée à partir de la composante nommée *IRmoy* qui correspond à la moyenne arithmétique des différentes composantes de l'image multi-spectrale. Cette méthode consiste à partager la dynamique de cette composante en intervalles égaux. Les pixels dont les niveaux de gris se situent dans un même intervalle sont affectés à la même classe. Le nombre d'intervalles est alors égal au nombre de classes  $K$  souhaitées [99, 100]. Après cette segmentation initiale, la technique markovienne est appliquée en prenant en considération les cinq composantes spectrales de l'image. La figure III.6 illustre un exemple des résultats de segmentation d'une image multi-spectrale MSG2. Les deux premières lignes dans cette figure fournissent les divers canaux infrarouges de l'image. Au-dessous, sont fournis les résultats de

la classification multi-composante en utilisant plusieurs formes d'énergie de Gibbs. Tous les résultats de cette figure sont donnés pour un nombre de classes  $K$  égale à 15 et pour un nombre d'itérations  $Itc$  égale à 10.  $Itc$  est le critère employé pour arrêter l'algorithme ICM. Les images des étiquettes sont affichées en couleurs afin de bien distinguer les différentes classes. La représentation des divers canaux est donnée en niveau de gris car la dynamique de ces derniers est de 256 niveaux.

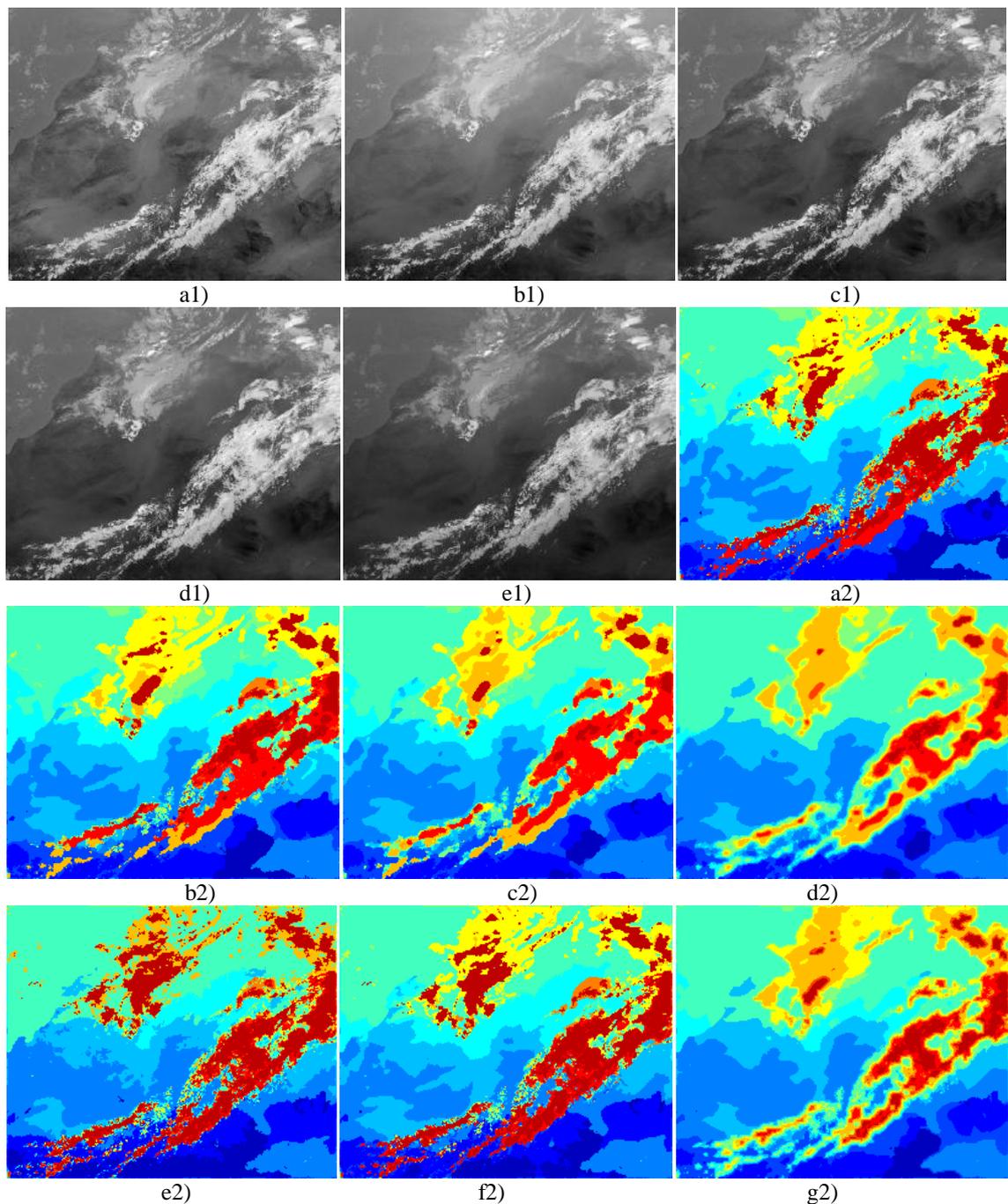


Fig. III.6 : Segmentation de l'image multi-spectral du 16 Janvier 2007 à 12h00 ( $K=15$ ,  $Itc=10$ )

1) les composantes de l'image : a) IR8.7, b) IR9.7 c) IR10.8, d) IR12.0, e) IR13.4

2) Résultats de la segmentation en utilisant l'énergie : a)  $E1$  b)  $E2$  ( $p=1/2$ ) c)  $E2$  ( $p=1$ ) d)  $E2$  ( $p=2$ ) e)  $E3$   
f)  $E4$  g)  $E5$

Nous donnons dans le tableau 1, pour chaque résultat de segmentation, la valeur  $Cr$  de la fonction de Borsotti, le nombre de régions connexes  $NR$  détecté et le pourcentage des pixels ayant changés de classes à la dernière itération  $Pr$ . Rappelons que tous ces résultats sont donnés pour un nombre d'itérations de l'algorithme égal à 10.

**Tableau 1** : Résultats des tests pour 3 images

| Energies de Gibbs | Image 1*, $K=15$ |      |      | Image 2*, $K=15$ |      |      | Image 3*, $K=15$ |      |      |
|-------------------|------------------|------|------|------------------|------|------|------------------|------|------|
|                   | $Pr$             | $Cr$ | $NR$ | $Pr$             | $Cr$ | $NR$ | $Pr$             | $Cr$ | $NR$ |
| $E1$              | 2.90             | 758  | 933  | 4.38             | 339  | 1160 | 2.62             | 860  | 1215 |
| $E2$ $p=1/2$      | 3.48             | 959  | 857  | 4.54             | 350  | 1214 | 2.45             | 713  | 1076 |
| $E2$ $p=1$        | 3.81             | 1076 | 842  | 4.42             | 477  | 1267 | 2.70             | 664  | 901  |
| $E2$ $p=2$        | 4.13             | 1396 | 1331 | 4.73             | 983  | 1306 | 3.94             | 725  | 740  |
| $E3$              | 2.49             | 2054 | 1716 | 5.13             | 1339 | 3336 | 2.84             | 1252 | 2134 |
| $E4$              | 2.57             | 946  | 1078 | 5.48             | 580  | 2059 | 2.66             | 920  | 1401 |
| $E5$              | 3.78             | 1297 | 791  | 6.05             | 336  | 1261 | 4.13             | 758  | 674  |

\* images Meteosat du 16 Janvier 2007 à 12h00 (image1), 15 Decembre 2006 à 12h00 (image 2) et 15 Novembre 2006 à 12h00 (image 3)

$K$ : nombre de classes,

$Pr$ : Pourcentage de pixels ayant change de classes à la dernière (10<sup>th</sup>) itération de l'algorithme ICM,

$Cr$ : Valeur de la fonction de Borsotti,

$NR$ : Nombre de régions connexes détecté dans l'image

En observant les résultats affichés sur le tableau 1, nous notons que les valeurs obtenues pour la fonction de Borsotti sont appréciables parce qu'elles sont comparables à celles fournies par Borsotti [68]. En effet,  $Cr$  reste inférieur à 10 000. Il est nécessaire de noter que plus le nombre de canaux utilisés augmente, plus la valeur de la fonction de Borsotti sera importante pour la même qualité de segmentation, parce que la distance  $E_r$  (voir l'Eq. II.34) deviendra plus significative. Effectivement cette distance est calculée en ajoutant les distances euclidiennes entre les valeurs des pixels de l'image d'entrée et les valeurs multi-composantes affectées aux classes dans l'image segmentée. Ainsi plus il y a de composantes, plus il y aura de termes à ajouter dans la distance  $E_r$  et plus celle-ci sera importante. Considérant que la valeur de la fonction de Borsotti pour une bonne segmentation de l'image couleur, qui comporte trois composantes, ne dépasse pas 10 000, sachant que les images traitées dans ce chapitre sont sur cinq composantes et au vu des divers valeurs de  $Cr$  données dans le tableau 1, nous pouvons conclure que les résultats de la segmentation des images MSG2 par les champs de Markov sont relativement appréciables.

Les images des étiquettes obtenues en utilisant une classification multi-composante (Fig. III.6) sont différentes quand nous utilisons différentes énergies de Gibbs, mais les nuages sont relativement bien détectés et les nuages de différentes densités sont bien séparés

dans la totalité des résultats (ce sont les classes qui sont représentées dans des couleurs oranges et rouges).

Dans ce travail, sept fonctions de Gibbs sont testées, parmi elles nous avons proposé deux, E2 (Eq. III.18) avec  $p=1/2$ , et E5 (Eq. III.21). Il n'est pas possible de choisir la meilleure fonction d'énergie parce que celle qui fournit le meilleur résultat est différente d'une image à l'autre. La fonction d'énergie E3 (Eq. III.9) donne les plus mauvais résultats dans tous les tests que nous avons réalisés, ce qui peut être clairement observé dans le tableau 1. L'énergie proposée E2 ( $p=1/2$ ) est toujours parmi les trois meilleurs résultats.

Le nombre d'itérations de l'algorithme ICM est fixé à 10. Ce nombre est largement suffisant parce que la valeur maximale que nous avons trouvée pour le taux  $Pr$  après la 10<sup>ème</sup> itération est de 6.05% (elle est obtenue pour l'image du 15 décembre 2006), ce qui est suffisant puisqu'elle est restée inférieure à 10%. Pour voir l'évolution de  $Pr$  en fonction des itérations, nous affichons en figure III.7, la moyenne arithmétique de ce taux, par énergie, quand la segmentation est effectuée sur plusieurs images.

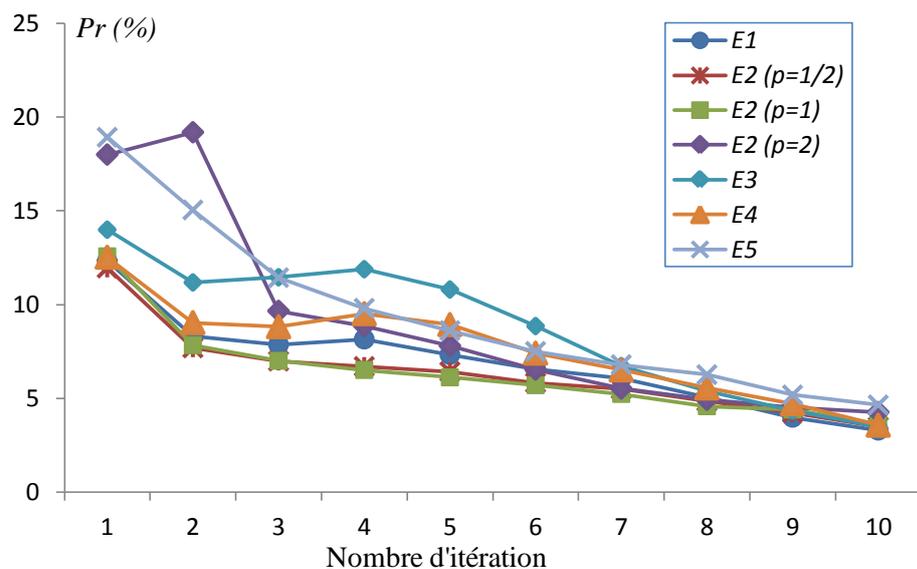


Fig. III.7 : Variation du taux de pixels ayant changé de classes à la  $i^{\text{ème}}$  itération de l'algorithme ICM

La convergence est rapide puisque dans tous les cas, après la troisième itération, le  $Pr$  obtenu est inférieur à 10%. A partir de la 9<sup>ème</sup> itération, dans la majorité des cas  $Pr$  est inférieur à 5%. Globalement, les courbes de  $Pr$  décroissent; il se peut qu'elles aient un maximum local, mais après, elles décroissent continuellement.

Sur tous les exemples testés, l'algorithme ne diverge jamais, même pour un nombre important d'itérations. Il peut osciller légèrement et localement sur quelques itérations mais il

continue globalement à converger et la valeur de  $Pr$  continue à diminuer. Pour avoir une idée sur l'évolution et les modifications subit au fur et à mesure des itérations, la figure III.8 fournit les résultats de la segmentation de l'image multi-spectrale du 16 janvier 2007 en utilisant l'énergie de Gibbs  $EI$  (Eq. III.16 et III.17).

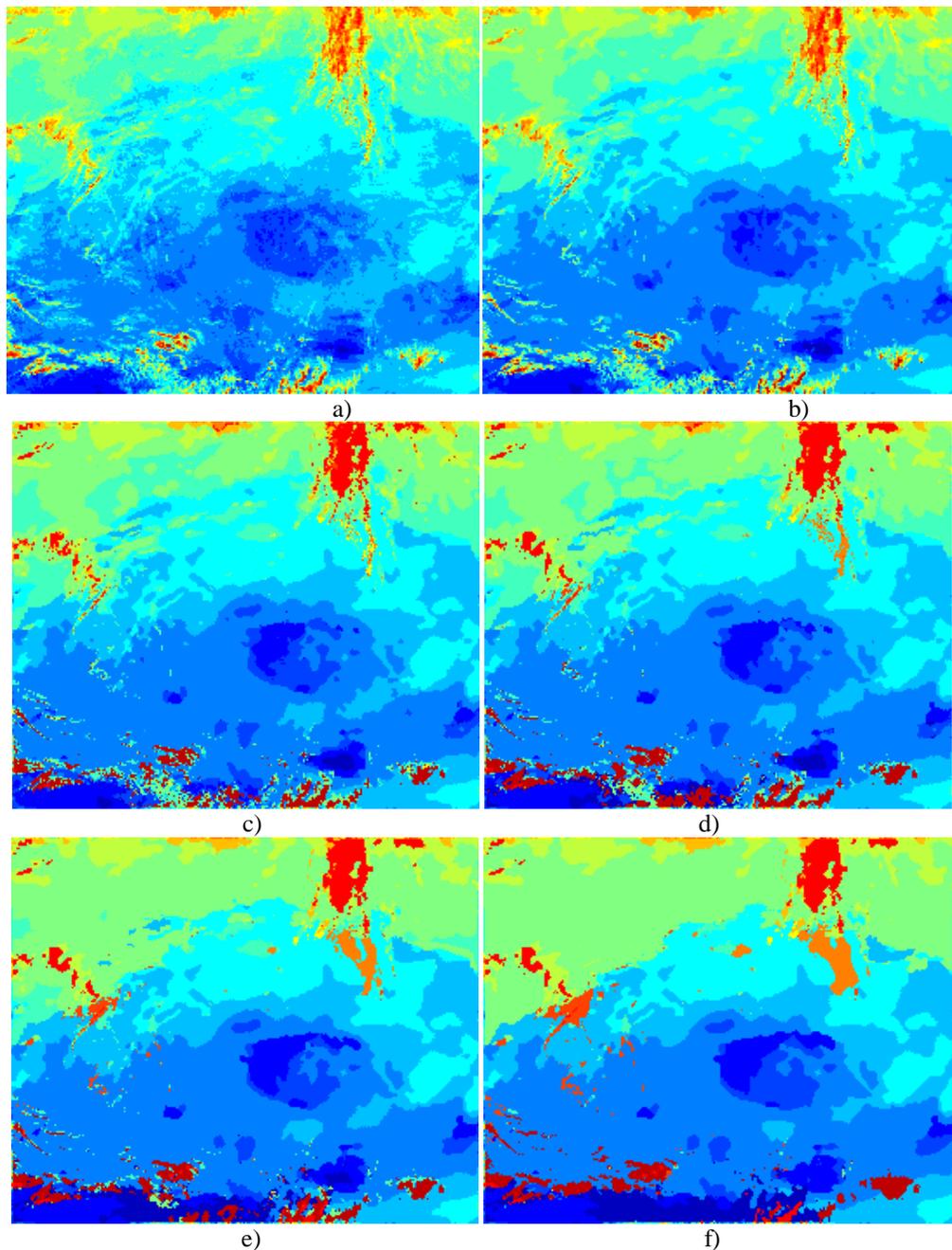


Fig. III.8 : Evolution des résultats de la segmentation de l'image du 16 décembre 2007 en fonction du nombre d'itération, pour  $K=15$  et pour l'énergie  $EI$   
a) classification initial b) 2 c) 5 d) 7 e) 10 et f) 20 itérations

Dans cette dernière figure, nous pouvons constater que plus le nombre d'itérations augmente plus les régions petites et isolées sont éliminées. Ce résultat est prévisible car, dans la

segmentation markovienne, pour classer un pixel il faut prendre en considération les informations de son voisinage. Cette caractéristique est appréciée lorsque les images traitées sont réelles. En effet, les phénomènes atmosphériques sont rarement isolés et restreints étant donné qu'il n'existe pas de variations brusques dans la nature. Nous avons testé le cas de 20 itérations pour la segmentation multi-spectrale de l'image du 16 janvier 2007. A la 20<sup>ème</sup> itération, le résultat trouvé est  $Pr=0.54\%$  et  $Cr=749$  contre  $Pr=2.9\%$  et  $Cr=758$  à la 10<sup>ème</sup> itération. Après la 10<sup>ème</sup> itération, nous ne constatons pas de modifications importantes dans les résultats de la segmentation. Cela est clairement visible sur la figure III.8 où les résultats de la segmentation à la 10<sup>ème</sup> et de la 20<sup>ème</sup> itération sont affichés.

## 5 Conclusion

Dans ce chapitre, nous avons présenté une méthode de segmentation d'images multi-spectrales MSG2 basée sur le formalisme markovien. Dans cette approche où les interactions spatiales et spectrales entre les pixels sont prises en compte, la classification des pixels est orientée au sens de la maximisation a posteriori (MAP). Pour obtenir la classification la plus probable en utilisant le MAP, il est nécessaire de modéliser le champ des observations (processus image) et le champ des classes (processus étiquette).

Le processus étiquette est modélisé en utilisant l'énergie de Gibbs. Sept formes d'énergie de Gibbs sont comparées dans le cadre de la segmentation multi-spectral. Deux d'entre elles sont nouvellement proposées dans ce travail. Cette expérimentation n'a pas été totalement exhaustive. En effet, le nombre de possibilités pour le choix de celles-ci est très important, il aurait été impossible de les examiner toutes. Seules les plus connues ou celles qui n'ont pas été appliquées à la segmentation des images multi-spectrales ont ici été utilisées. Les résultats montrent que le choix de la fonction d'énergie de Gibbs influe sur la qualité de la segmentation, bien que cette influence ne soit pas très significative. Les deux fonctions que nous avons proposées donnent, de manière générale, de bons résultats.

Comme dans le cas du chapitre précédent, les résultats sont évalués avec la fonction de Borsotti qui est habituellement connue pour être fiable. Les différents essais prouvent que les résultats sont satisfaisants. En effet, l'algorithme converge en un nombre restreint d'itérations, les valeurs de la fonction de Borsotti sont acceptables et les nuages sont bien détectés et séparés.

## Chapitre IV

# Segmentation markovienne d'images couleur texturées en utilisant des paramètres basés sur la classification spectrale

### 1 Introduction

La couleur et la texture sont des caractéristiques essentielles puisqu'elles sont largement observées dans la plupart des images. La texture caractérise des variations locales de l'intensité dans l'image ; elle est définie par un ensemble de paramètres qui décrivent les relations spatiales entre les pixels voisins [101, 102]. La couleur est un attribut intrinsèque d'une image, elle fournit plus d'informations qu'une seule valeur d'intensité. Dans le domaine du traitement d'image, la couleur et la texture sont plutôt considérées séparément. Cependant l'association de ces deux types de caractéristiques peut mener à des méthodes d'analyse puissantes. Dans ce cas-ci, nous pouvons considérer une texture-couleur comme une configuration décrite par le rapport entre sa distribution chromatique et sa distribution structurale. Deux images comprenant la même couleur mais différentes configurations de texture ou la même configuration de texture mais différentes couleurs sont deux texture-couleurs différentes [103].

La segmentation d'images couleur texturées est souvent réalisée en deux étapes. La première étape consiste à extraire les paramètres de texture et de couleur, la deuxième étape à regrouper ces paramètres dans différentes classes. Les diverses méthodes de segmentation des images couleur texturées proposées dans la littérature diffèrent principalement dans l'approche employée pour extraire les paramètres de texture et/ou dans l'algorithme utilisé pour regrouper les pixels similaires.

Dans ce travail, nous suivons la même démarche. La méthode de classification utilisée est basée sur les champs aléatoire de Markov (MRF) par contre l'extraction des paramètres de texture est réalisé avec une nouvelle technique que nous proposons [16] basée sur les marches aléatoires sur graphes (random walk in graph) qui est déduite du domaine de la classification spectrale. La classification spectrale est un algorithme de classification qui résulte des concepts de la théorie des graphes. Dans ce travail nous proposons d'utiliser la classification

spectrale non pas en tant que technique de classification mais comme une méthode de calcul des paramètres de texture.

## **2 Etat de l'art de la segmentation d'images couleur texturées**

Dans la littérature, quelques travaux traitent de la segmentation des images couleur texturées. Les méthodes proposées sont généralement déclinées en deux étapes. Dans la première étape, les paramètres de texture et de couleur sont extraits séparément ou conjointement en utilisant les techniques citées précédemment. Dans la deuxième étape, ces paramètres sont regroupés dans différentes classes, en utilisant généralement des méthodes de classification non supervisées.

Panjwani et Healy [88] ont présenté un algorithme de segmentation qui utilise les champs aléatoires de Markov pour segmenter des images couleurs texturées. L'interaction spatiale est décrite par et entre les bandes RGB de l'image couleur. Le modèle développé décrit une texture-couleur en termes d'une dépendance statistique du vecteur RGB mesuré en un pixel par rapport au vecteurs RGB des pixels voisins. L'algorithme de classification est contrôlé par un critère déduit de la vraisemblance de l'image.

Eom [101] utilise une approche basée sur le modèle des moyennes mobiles 2D afin d'extraire les paramètres de texture. Un algorithme de classification floue utilisant ces paramètres est exploité pour la segmentation non supervisée.

Deng et Manjunath [104] ont proposé une nouvelle approche dénommée JSEG pour la segmentation non supervisée des images couleur texturées. Cette méthode se compose de deux étapes indépendantes : quantification de couleur et segmentation spatiale. Dans la première phase, les couleurs de l'image sont quantifiées en un ensemble de classes représentatives. Les valeurs des pixels de l'image sont alors remplacées par les étiquettes correspondantes aux classes des couleurs, formant de ce fait une carte des classes de l'image. Dans la seconde étape, la segmentation spéciale est exécutée directement sur cette carte des classes sans considérer les couleurs des différents pixels.

Haindel et Mikes [105] ont utilisé une approche basée sur la modélisation par les champs aléatoires autorégressifs causals (Causal Autoregressive Random field (CAR)) pour l'extraction des paramètres de texture. L'algorithme de classification est l'algorithme EM modifié.

Kato et al. [106] ont proposé une méthode de segmentation d'images couleur texturées basées sur les modèles MRF multicouches et l'analyse par des mélanges de gaussiennes. Kato

et Pong [98] ont proposé une autre segmentation MRF en combinant des paramètres de couleur et de texture. Pour extraire les paramètres de texture ils ont exploité les filtres multicanaux de Gabor (4 filtres). Deux méthodes différentes d'optimisation, à savoir le recuit simulé et l'ICM (Iterated Conditional Modes) ont été comparées.

Allili et Ziou [107] proposent une méthode automatique pour la segmentation d'images couleur texturées basée sur les contours actifs combinant les informations régions et contours. L'information région est basée sur le mélange de modèles combinant les paramètres de couleur et de texture. Les paramètres de texture sont calculés à partir du corrélogramme du voisinage de chaque pixel et les paramètres de couleur utilisés sont les composantes couleurs CIELab.

Shi et Funt [108] emploient une représentation de type quaternion pour les couleurs RGB afin de segmenter des images couleur texturées. Un ensemble d'échantillons de texture est extrait de l'image en utilisant quelques sous-fenêtres carrées. Les valeurs RGB des pixels situées dans chaque fenêtre sont représentées comme des quaternions formant ainsi les colonnes des vecteurs. L'ensemble d'apprentissage est alors représenté comme une matrice. Celle-ci est décomposée en valeurs singulières et vecteurs propres de quaternion. Ces vecteurs de paramètres quaternion sont les entrées de l'algorithme k-means. La sortie de cet algorithme fournit les centroïdes décrivant les moyennes des différentes classes de texture. Au final, on assigne à chaque pixel l'étiquette de l'ensemble d'apprentissage dont le centroïde est le plus proche en termes de distance euclidienne.

Islam et al. [109] proposent une segmentation non supervisée d'images couleur texturées en utilisant les modèles SVFMM (Spatially Variant Finite Mixture Model) et MRF. Le nombre de classes est déterminé en utilisant une approche par "Clusters Ensembles". Les paramètres de texture sont extraits au moyen de la transformée en ondelettes Haar et les paramètres couleur choisis sont CIELuv.

Kim et Hong [110] formulent la segmentation d'images couleur texturées en termes de minimisation d'énergie sur les coupes de graphe "graph cuts". Les paramètres couleur et texture sont modélisés par des mélanges finis de modèles multi-variables avec un nombre inconnu de composantes. Le vecteur des paramètres couleur se compose des paramètres RGB. Le procédé de génération des paramètres de texture est basé sur un banc de filtres de Gabor. L'algorithme de classification est l'algorithme K-means.

Brox et al. [111] utilisent la couleur, la texture et le mouvement dans la segmentation et la poursuite (tracking) en se basant sur les "level sets". Ils utilisent le concept des tenseurs non

linéaires afin d'extraire les paramètres de texture et les combinent ensuite avec les paramètres de couleur et de mouvement.

### 3 Classification spectral (Spectral clustering)

La classification spectrale est initialement introduite par Donath et Hoffman [112], qui furent les premiers à suggérer la construction des partitions de graphe basées sur les vecteurs propres des matrices d'adjacence. Fiedler [113] suggère ensuite d'employer le deuxième vecteur propre du graphe Laplacien pour partitionner le graphe. Depuis, beaucoup de travaux traitent de la classification spectrale [114-117].

La classification spectrale est un algorithme de classification qui résulte des concepts de la théorie des graphes. L'idée de base est la construction d'un graphe pondéré à partir de l'ensemble de données initial où chaque nœud représente un échantillon et chaque arête pondérée tient compte de la similarité entre deux échantillons. De nouveaux échantillons sont alors construits en calculant la décomposition en valeur propre de la matrice de Laplace du graphe pondéré, transformant ainsi la représentation des données initiales. Après cela, l'algorithme de classification K-means est utilisé sur ces nouvelles données afin de séparer les différentes classes [116,117].

#### 3.1 Formulation de la classification en utilisant les graphes

Classer un ensemble de données consiste à le subdiviser en plusieurs groupes tels que les données se trouvant dans le même groupe sont similaires et les données se trouvant dans des groupes différents sont dissimilaires entre eux.

Soit un ensemble de  $n$  objets  $V = \{v_1, v_2, \dots, v_n\}$  à classer où chaque objet est représenté par un vecteur de  $D$  paramètres. Pour réaliser sa classification, nous pouvons le formaliser sous la forme d'un graphe de similarité  $G$  tel que :

$$G = (V, E, W) \quad (\text{IV.1})$$

où  $V$  est l'ensemble des nœuds  $v_i$  représentant l'ensemble des objets et  $E$  est l'ensemble des arêtes. Chaque arête entre deux nœuds  $v_i$  et  $v_j$  est affecté d'un poids  $w_{ij}$ . La matrice d'adjacence pondérée du graphe est alors la matrice  $W = (w_{ij})_{i,j=1, \dots, n}$ .

Le problème de la classification peut être reformulé en utilisant le graphe de similarité de la manière suivante: il s'agit de trouver une partition du graphe  $A_1, \dots, A_K$  tels que les arêtes entre les différentes classes possèdent une faible pondération (ce qui signifie que les données dans des différentes classes sont dissemblables les unes des autres) et les arêtes à l'intérieur

d'une classe aient un poids élevé (ce qui signifie que les données dans une même classe sont similaires les unes aux autres) [116]. Pour être en mesure de formaliser cette intuition, il est nécessaire de rappeler quelques notions de base liées aux graphes qui seront indispensables pour l'étude de la classification spectrale.

### 3.2 Notions de base sur les graphes

Il existe trois types de graphes de similarité [116]:

- Le graphe de  $\varepsilon$ -voisinage ( $\varepsilon$ -neighborhood graph) dans lequel tous les couples de nœuds dont la distance est plus petite que  $\varepsilon$  sont connectés. C'est un graphe non orienté.
- Le graphe de  $k$ -plus proches voisins ( $k$ -nearest neighbor graph) dans lequel le nœud  $v_i$  est connecté au nœud  $v_j$  si  $v_j$  est parmi les  $k$  plus proches voisins de  $v_i$ . C'est un graphe orienté. Il existe deux méthodes pour rendre ce graphe non orienté. La première consiste à ignorer les directions des arêtes, le graphe qui en résulte est alors dit le graphe des  $k$  plus proches voisins standard ( $k$ -nearest neighbor standard graph). La deuxième méthode consiste à relier les nœuds  $v_i$  et  $v_j$  si  $v_i$  est parmi les  $k$  plus proches voisins de  $v_j$  et le  $v_j$  est parmi les  $k$  plus proches voisins de  $v_i$ . Ce graphe est alors dit graphe de  $k$ -plus proche voisin mutuel (mutual  $k$ -nearest neighbor graph).
- Le graphe totalement connecté (fully connected graph) où tous les nœuds sont connectés entre eux avec une similarité positive et toutes les arêtes sont affectées d'un poids  $w_{ij}$ .

Dans la littérature, plusieurs techniques permettent d'obtenir une mesure de similarité entre deux objets. Le choix de la fonction de similarité dépend essentiellement du domaine de provenance des données (par exemple, fouille de documents, fouille de données web, etc.), mais également du type de données (qui peuvent être décrit par des variables numériques, catégorielles, binaires, ...etc). Cependant, la formule la plus utilisée est celle du noyau Gaussien décrite par [118]:

$$w_{ij} = w(v_i, v_j) = \exp\left(-\frac{\|v_i - v_j\|}{2\sigma^2}\right) \quad (\text{IV. 2})$$

où  $\| \cdot \|$  désigne une mesure de distance et  $\sigma$  un paramètre d'échelle qui contrôle la largeur du voisinage.

Le degré d'un nœud  $v_i \in V$  est défini par :

$$d_i = \sum_{j=1}^n w_{ij} \quad (\text{IV. 3})$$

La matrice degré  $\mathcal{D}$  est définie comme étant la matrice diagonale avec les degrés  $d_1, d_2, \dots, d_n$  sur la diagonale.

### 3.3 Matrices laplaciennes d'un graphe

La classification spectrale est principalement fondée sur l'étude des matrices laplaciennes d'un graphe. Il existe tout un champ dédié à l'étude de ces matrices, appelé "théorie spectrale des graphes" (voir [119, 120]).

Plusieurs matrices laplaciennes d'un graphe sont proposées dans la littérature. Cependant, ces matrices peuvent être normalisées ou non normalisées.

La matrice laplacienne non normalisée d'un graphe est définie par :

$$L = \mathcal{D} - W \quad (\text{IV. 4})$$

où  $\mathcal{D}$  est la matrice de degré et  $W$  la matrice de pondération.

$L$  est symétrique et semi-définie positive. Sa plus petite valeur propre est égale à 0. Elle possède  $n$  valeurs propres réelles non-négatives,  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . Les valeurs propres et vecteurs propres de la matrice laplacienne non normalisée d'un graphe peuvent être utilisés pour décrire de nombreuses propriétés des graphes (voir [120, 121]). Le nombre  $k$  de valeur propre nuls de  $L$  correspond au nombre de composantes connexes  $A_1, \dots, A_k$  dans le graphe. Pour rappel,  $A_i$  est une composante connexe si deux nœuds de  $A_i$  peuvent être joints par un chemin tels que tous les nœuds de ce chemin se situent également dans  $A_i$  et s'il n'y a aucune connexion entre les nœuds de  $A_i$  et  $A_{j \neq i}$ .

On trouve dans la littérature principalement deux matrices laplaciennes normalisées d'un graphe. Ces deux matrices sont étroitement liées l'une à l'autre et sont définies comme suit :

$$L_{sym} = \mathcal{D}^{-1/2} L \mathcal{D}^{-1/2} = I - \mathcal{D}^{-1/2} W \mathcal{D}^{-1/2} \quad (\text{IV. 5})$$

$$L_{rw} = \mathcal{D}^{-1} L = I - \mathcal{D}^{-1} W \quad (\text{IV. 6})$$

On note la première matrice par  $L_{sym}$  car c'est une matrice symétrique et la seconde par  $L_{rw}$  car elle est étroitement liée à la marche aléatoire sur graphe (random walk in graph) [119].  $L_{sym}$  et  $L_{rw}$  sont semi-définies positives et possèdent  $n$  valeurs propres réelles non négatives  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ .

$\lambda$  est une valeur propre de  $L_{rw}$  de vecteur propre  $\psi$  si et seulement si  $\lambda$  est une valeur propre de  $L_{sym}$  de vecteur propre  $\omega = \mathcal{D}^{1/2} \psi$ .

$\lambda$  est une valeur propre de  $L_{rw}$  de vecteur propre  $\psi$  si et seulement si  $\psi$  et  $\lambda$  résolvent le problème propre généralisé  $L\psi = \lambda D\psi$ .

Comme dans le cas de la matrice laplacienne non normalisée, le nombre de valeur propre 0 de la matrice laplacienne normalisée correspond au nombre des composantes connexes du graphe. Plus de détails sur les propriétés des matrices laplaciennes normalisées peuvent être trouvés dans [119].

### 3.4 Algorithmes de classification spectrale

Les algorithmes de classification spectrale calculent les  $k$  premiers vecteurs propres  $\psi_1, \psi_2, \dots, \psi_k$  correspondant aux  $k$  plus petites valeurs propres. Ensuite une nouvelle matrice  $U \in \mathbb{R}^{n \times k}$  est construite, contenant les vecteurs propres en colonnes. Les vecteurs  $v_1, v_2, \dots, v_n$  correspondant aux lignes de  $U$  sont interprétés en tant que nouveaux points d'informations. Le but principal de ces algorithmes est d'opérer un changement d'espaces de représentation des données. Cette modification de représentation améliore les propriétés des classes, de sorte que des méthodes de regroupement simple n'aient aucune difficulté à détecter les classes dans la nouvelle représentation [116].

Les trois algorithmes de classification spectrale sont indiqués sur les figures IV.1, IV.2 et IV.3. Ils sont plutôt similaires, mais à part qu'ils utilisent trois types de matrices laplaciennes. L'algorithme de classification spectrale utilisant la matrice laplacienne non normalisée est donné sur la figure IV.1.

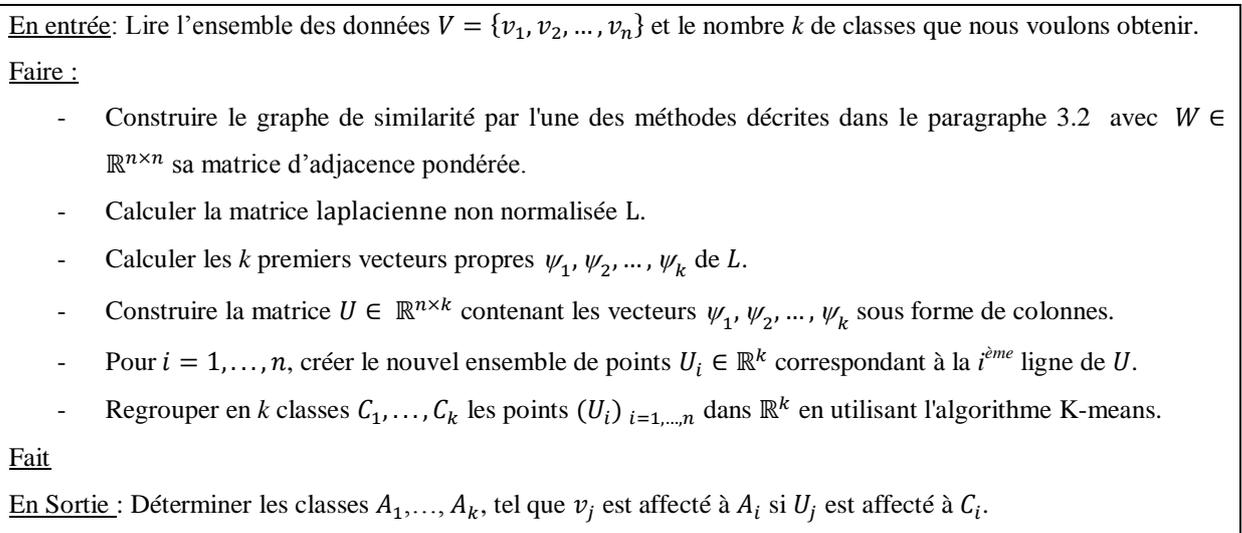


Fig. IV.1 : Algorithme de classification spectrale non normalisée

Il existe deux versions différentes de classification spectrale normalisée selon la matrice laplacienne utilisée qui sont illustrées sur les figures IV.2 et IV.3.

|   |
|---|
| <p><u>En entrée:</u> Lire l'ensemble des données <math>V = \{v_1, v_2, \dots, v_n\}</math> et le nombre <math>k</math> de classes que nous voulons obtenir.</p> <p><u>Faire :</u></p> <ul style="list-style-type: none"> <li>- Construire le graphe de similarité par l'une des méthodes décrites dans le paragraphe 3.2.</li> <li>- Calculer sa matrice d'adjacence pondérée <math>W \in \mathbb{R}^{n \times n}</math>.</li> <li>- Calculer la matrice laplacienne non normalisée <math>L</math>.</li> <li>- Calculer les <math>k</math> premiers vecteurs propres <math>\psi_1, \psi_2, \dots, \psi_k</math> du problème propre généralisé <math>L\psi = \lambda D\psi</math>.</li> <li>- Soit la matrice <math>U \in \mathbb{R}^{n \times k}</math> contenant les vecteurs <math>\psi_1, \psi_2, \dots, \psi_k</math> sous forme de colonnes.</li> <li>- Pour <math>i = 1, \dots, n</math>, créer le nouvel ensemble de points <math>U_i \in \mathbb{R}^k</math> correspondant à la <math>i^{\text{ème}}</math> ligne de <math>U</math>.</li> <li>- Regrouper en classes <math>C_1, \dots, C_k</math> les points <math>(U_i)_{i=1, \dots, n}</math> dans <math>\mathbb{R}^k</math> avec l'algorithme K-means.</li> </ul> <p><u>Fait</u></p> <p><u>En Sortie:</u> déterminer les classes <math>A_1, \dots, A_k</math>, tel que <math>v_j</math> est affecté à <math>A_i</math> si <math>U_j</math> est affecté à <math>C_i</math>.</p> |
|---|

Fig. IV.2 : Algorithme de classification spectrale normalisée selon Shi et Malik [122]

Notez que l'algorithme de la figure IV.2 utilise les vecteurs propres généralisés de  $L$ , qui correspondent aux vecteurs propres de la matrice  $L_{rw}$ . Donc, en fait, l'algorithme fonctionne avec les vecteurs propres de la matrice laplacienne normalisée  $L_{rw}$ , d'où son nom classification spectrale normalisée. L'algorithme suivant utilise également une laplacienne normalisée, mais cette fois c'est la matrice  $L_{sym}$  qui est utilisée au lieu de  $L_{rw}$ .

|  |
|--|
| <p><u>En entrée:</u> Lire l'ensemble des données <math>V = \{v_1, v_2, \dots, v_n\}</math> et le nombre <math>k</math> de classes que nous voulons obtenir.</p> <p><u>Faire :</u></p> <ul style="list-style-type: none"> <li>- Construire le graphe de similarité par l'une des méthodes décrites dans le paragraphe 3.2.</li> <li>- Calculer sa matrice d'adjacence pondérée <math>W \in \mathbb{R}^{n \times n}</math>.</li> <li>- Calculer la matrice laplacienne normalisée <math>L_{sym}</math>.</li> <li>- Calculer les <math>k</math> premiers vecteurs propres <math>\psi_1, \psi_2, \dots, \psi_k</math> de <math>L_{sym}</math>.</li> <li>- Soit <math>U \in \mathbb{R}^{n \times k}</math> la matrice contenant les vecteurs <math>\psi_1, \psi_2, \dots, \psi_k</math> sous forme de colonnes.</li> <li>- Formuler la matrice <math>T \in \mathbb{R}^{n \times k}</math> en normalisant les lignes de <math>U</math>, de la manière suivante :</li> </ul> $t_{ij} = \frac{U_{ij}}{(\sum_{j=1}^k U_{ij}^2)^{1/2}} \quad (IV.7)$ <ul style="list-style-type: none"> <li>- Pour <math>i = 1, \dots, n</math>, créer le nouvel ensemble de points <math>t_i \in \mathbb{R}^k</math> correspondant à la <math>i^{\text{ème}}</math> ligne de <math>T</math>.</li> <li>- Regrouper en classes <math>C_1, \dots, C_k</math> les points <math>(t_i)_{i=1, \dots, n}</math> dans <math>\mathbb{R}^k</math> avec l'algorithme K-means.</li> </ul> <p><u>Fait</u></p> <p><u>En sortie:</u> déterminer les classes <math>A_1, \dots, A_k</math>, tel que <math>v_j</math> est affecté à <math>A_i</math> si <math>U_j</math> est affecté à <math>C_i</math>.</p> |
|--|

Fig. IV.3 : Algorithme de classification spectrale normalisée selon Ng, Jordan et Weiss [123]

#### 4 Formalisation de la méthode de segmentation proposée

Nous proposons dans ce chapitre de segmenter des images couleur texturées en regroupant les pixels ayant des paramètres couleur et/ou texture similaires dans les mêmes classes. On se propose de tester la segmentation d'images couleur texturées en utilisant seulement les paramètres couleur, puis en utilisant seulement les paramètres texture et enfin en combinant les paramètres couleur avec les paramètres texture. Chaque pixel de l'image sera donc décrit par un vecteur de paramètres regroupant les paramètres couleur et les paramètres de texture. Avant de décrire l'extraction de ces paramètres, nous définissons d'abord le formalisme utilisé.

L'idée dans ce formalisme est de représenter une image par un graphe pondéré et non orienté  $G=(E,V,W)$ , avec  $V = (v_1, v_2, \dots, v_n)$  qui est un ensemble fini de nœuds représentant les pixels de l'image et  $E$  un ensemble fini d'arrêtes pondérés par la similarité  $W$  entre les nœuds connectés. Soit  $f(v)$ , une fonction définie sur le nœud  $v$  dans un espace à  $D$ -dimension, et représentée par le vecteur  $\{f_1^v, f_2^v, f_3^v, \dots, f_D^v\} \in \mathfrak{R}^D$ . Nous noterons  $u \sim v$  pour signifier que le nœud  $u$  appartient à  $\varepsilon$ -voisinage ( $\varepsilon$ -neighborhood) du nœud  $v$  ( $u \in N_\varepsilon(v)$ ), qui est défini par:

$$N_\varepsilon(v) = \left\{ u \in V, f(u) = (f_1^u, f_2^u, \dots, f_D^u) / \sum |f_i^u - f_i^v| \leq \varepsilon, 1 \leq i \leq D \right\} \quad (\text{VI.8})$$

$N_\varepsilon(v)$  comprend tous les nœuds proches du nœud  $v$  (voir Fig. IV.4). Nous notons  $g$  le nombre de ces nœuds.

Nous définissons ensuite une fonction patch  $F$  autour du nœud  $v$  comme étant l'ensemble des nœuds de son voisinage (voir Fig. IV.4). Cette fonction est donc définie comme suit :

$$F(v) = \left\{ F_i^v = f(u_i) / u_i \text{ est le nœud à la position } i \text{ dans le voisinage, } i = 1, \dots, h \right\} \quad (\text{VI.9})$$

avec  $h$  dépendant de la taille du voisinage.

La fonction de pondération  $w$ , appelée aussi noyau gaussien, donne une mesure de similarité entre un nœud et ses voisins. Elle peut intégrer des fonctionnalités localisées et non localisées:

$$w(u,v) = \begin{cases} \exp\left(-\frac{\|f(u) - f(v)\|^2}{\sigma_1^2}\right) \exp\left(-\frac{\|F(u) - F(v)\|^2}{\sigma_2^2}\right), & \text{pour chaque } u \sim v \\ 0 & \text{ailleurs} \end{cases} \quad (\text{VI.10})$$

Les paramètres d'échelle  $\sigma_i$  peuvent être estimés en utilisant les écarts type des fonctions  $\|f(u) - f(v)\|$  et  $\|F(u) - F(v)\|$  définies sur le graphe.

$f(v)$  représente le vecteur des paramètres du nœud  $v$ . Ce vecteur peut être composé de différents éléments qui peuvent être, par exemple, ses coordonnées, ses paramètres couleur, ses paramètres de texture, etc ...

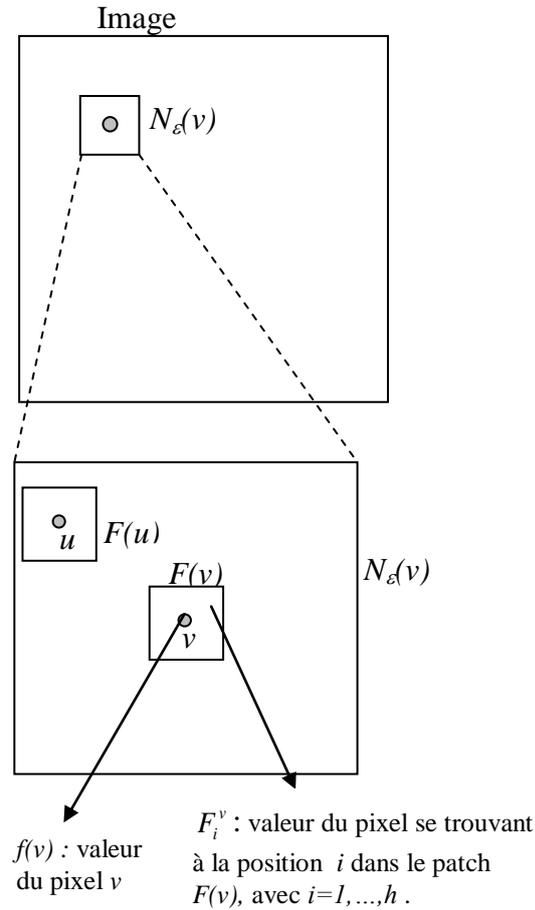


Fig. IV.4 : Représentation du voisinage  $N_d(v)$  et du patch  $F(v)$  définie pour le nœud  $v$

Dans notre formalisme  $f(v)$  sera constitué des paramètres couleur  $f_v^{col}$  et/ou des paramètres de texture  $f_v^{tex}$ . La dimension du vecteur  $f(v)$  dépend du test à réaliser.  $f(v) = f_v^{col} = (f_v^{c1}, f_v^{c2}, f_v^{c3})$  si la segmentation est réalisée en utilisant seulement les trois paramètres couleurs,  $f(v) = f_v^{tex} = (f_v^{tc1}, f_v^{tc2}, f_v^{tc3})$  dans les cas où seulement les trois paramètres de texture sont utilisés et  $f(v) = (f_v^{col}, f_v^{tex}) = (f_v^{c1}, f_v^{c2}, f_v^{c3}, f_v^{tc1}, f_v^{tc2}, f_v^{tc3})$  si la segmentation est réalisée en combinant les paramètres couleur avec les paramètres texture formant ainsi une image multi-protocole à six composantes. Dans les sections qui suivent, nous expliquons la manière dont les paramètres couleur et textures sont extraits. Le formalisme que nous venons de développer sera utilisé pour l'extraction des paramètres de texture.

## 5 Les paramètres couleur

L'extraction des paramètres couleurs est relativement simple et ces derniers se sont avérés pertinents dans le domaine de la segmentation des images couleur [98, 106].

L'image couleur initiale notée  $x$  est originalement représentée par ces paramètres couleur R (rouge), G (vert) et B (bleu). Les paramètres couleurs  $f_v^{col}$  du pixel  $v$  sont alors un vecteur à 3D dans l'espace de représentation RGB, tels que :

$$f_v^{col} = (f_v^{c1}, f_v^{c2}, f_v^{c3}) = (f_v^R, f_v^G, f_v^B) \quad (\text{IV.11})$$

où  $c_{i(i=1,2,3)}$  sont les composantes R, G et B.

D'autres espaces de couleurs, tels que les espaces HSV et le CIE L\*a\*b\* peuvent également être utilisés pour représenter une image couleur. Ces espaces sont fondamentalement différents de l'espace RGB car ils séparent l'information intensité (luminance) de l'information couleur (chrominance). Ils sont des conversions non-linéaires et réversibles de l'espace RGB. Plus d'informations sur ces espaces et d'autres espaces de représentation de l'image couleur peuvent être trouvées au paragraphe I.5 et dans [24].

## 6 Les paramètres de texture

La texture caractérise des variations locales de l'intensité dans l'image ; elle est définie par un ensemble de paramètres qui décrivent les relations spatiales entre les pixels voisins [101, 102]. De nombreuses approches pour l'extraction des paramètres de texture basée sur les modèles statistiques d'image [124], les modèles des champs aléatoires de Markov [125, 126], les transformation par ondelettes [127, 128], la dimension de fractale [129], la matrice de Cooccurrence de niveau de gris [130, 131], les fonctions de Gabor [132, 133], les configuration binaire locale (*local binary pattern*) [69], les transformation en valeur propres [134] ont été développées.

Nous proposons dans cette section un nouveau descripteur de texture en utilisant la théorie des graphes et la classification spectrale. Chaque pixel de l'image  $v$  sera caractérisé par un vecteur de paramètres de textures  $f_v^{tex}$ . Ce nouveau descripteur est basé sur une représentation par graphe de l'ensemble des données multidimensionnelles de l'image. Nous définissons une fonction de poids  $w$  qui reflète le degré de similitude entre chaque paire de nœuds. Le graphe  $G$  est associé donc à sa matrice  $W$  de similarité.  $W(u, v) = w(u, v)$  si les nœuds  $v$  et  $u$  sont connectés et  $W(u, v) = 0$  sinon.

La fonction poids incorpore seulement les variations non localisées parce que les descripteurs que l'on souhaite extraire dans cette section sont des descripteurs de texture. La fonction de similarité utilisée est alors identique à l'Eq. IV.10 sans les variations localisées :

$$w(u, v) = \exp\left(-\|F(v) - F(u)\|^2 / 2\sigma^2\right) \quad (\text{IV.12})$$

La fonction de pondération  $w$  tient en compte la distance entre deux patches connectés  $F(u)$  et  $F(v)$  qui sont dans le même voisinage  $N_\delta(v)$ .  $F_i^v$  représente la valeur du pixel situé à la position  $i$  dans le patch  $F(v)$ , avec  $i = 1, \dots, h$ . La distance entre les patches peut s'exprimer sous forme d'une distance euclidienne donnée par :

$$\|F(v) - F(u)\| = \sqrt{\sum_{i=1}^h (F_i^v - F_i^u)^2} \quad (\text{IV.12})$$

Rappelons que le degré  $d(v)$  du nœud  $v$  est définie par:

$$d(v) = \sum_{u \sim v} w(u, v) \quad (\text{IV.13})$$

La matrice degré  $\mathcal{D}$  est alors définie comme une matrice diagonale avec les degrés  $d_1, \dots, d_g$  sur la diagonale.

$$\mathcal{D} = \begin{bmatrix} d_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & d_g \end{bmatrix} \quad (\text{IV.14})$$

A partir des matrices de similitude et des degrés déterminées au voisinage  $N_\delta(v)$ , une matrice  $T$  de transition peut être déduite en utilisant les marches aléatoires sur graphe.

Une marche aléatoire sur graphe est un processus stochastique qui permet des sauts aléatoirement d'un nœud à un autre. Lorsque le graphe est pondéré, le déplacement vers un voisin s'opère avec une probabilité proportionnelle au poids de l'arête correspondante. Plus de détails sur les marches aléatoires et marche aléatoire sur graphe peuvent être trouvés dans [135-138].

Formellement, la probabilité de passage du nœud  $u$  au nœud  $v$  en une seule étape est donnée par :

$$p^{(1)}(u, v) = w(u, v) / d(u) \quad (\text{IV.15})$$

La matrice de transition  $T$  de  $G$  est donnée par:

$$T = \{p^{(1)}(u, v) / u, v \in V, u \sim v\} \quad (\text{IV.16})$$

Sous forme matricielle,  $T$  s'écrit :

$$T = \mathcal{D}^{-1}W \quad (\text{IV.17})$$

$T$  est matrice asymétrique, diagonalisable et sa décomposition en valeur et vecteurs propres fournit un ensemble  $\{\lambda^i, \psi^i\}$  tel que les valeurs ses propres  $\{\lambda^i\}$  sont comprises entre 0 et 1 ( $I = \lambda^1 \geq \lambda^2 \geq \dots \lambda^g \geq 0$ ) et  $\{\psi^i\}$  sont leurs vecteurs propres associés.

A des fins de classification, la décomposition spectrale de  $T$  est utilisée pour générer une représentation de faible dimension des nœuds du graph  $G$  à l'instant  $t$  où pour chaque nœud ces coordonnées sont données par:

$$\Psi_t(u) = \{\lambda_1^1 \psi^1(u), \lambda_1^2 \psi^2(u), \dots, \lambda_1^g \psi^g(u)\}^T \quad (\text{IV.18})$$

Cette transformation est communément utilisée pour la réduction des données et des dimensions. Elle permet de passer d'un espace à  $n$ -dimensions vers un espace euclidien réduit à  $m$  dimensions et de représenter l'ensemble des informations sous forme de propriétés structurelles du graphe (voir pour détail [136, 138]).

Puisque la somme des valeurs propres de  $T$  correspond à sa trace, nous utiliserons cette dernière pour caractériser le pixel  $v$  dans un espace couleur  $c$ .

$$f_v^{tc} = \sum_{i=1}^g \lambda_i = \sum_{i=1}^g T_{ii} \quad (\text{IV.19})$$

Au final, la caractéristique  $f_v^{tc}$  peut être considéré comme un paramètre de texture car elle est calculée en tenant compte non seulement du pixel seule, mais aussi en considérant son large voisinage. Il faut noter que les voisinages et les patches se chevauchent.

Dans le cas des images couleur, la procédure d'extraction des paramètres de texture sera appliqué séparément à chaque composante couleur noté  $c1$ ,  $c2$  et  $c3$ . Ainsi, chaque pixel  $v$  de l'image couleur, sera caractérisé par un vecteur 3D de paramètres de texture :

$$f_v^{tex} = (f_v^{tc1}, f_v^{tc2}, f_v^{tc3}) \quad (\text{IV.20})$$

où  $f_v^{tc1}$ ,  $f_v^{tc2}$  et  $f_v^{tc3}$  sont les paramètres de texture du pixel  $v$  déterminée en utilisant l'équation IV.19 sur les composantes couleur  $c1, c2$  et  $c3$  respectivement.

Un exemple donnant une image de texture composée par les paramètres  $\{f_v^{tex}\}$  est présenté en figure IV.5. La taille des patches  $h$  est  $(7 \times 7)$  et la taille du voisinage  $g$  est  $(15 \times 15)$ .

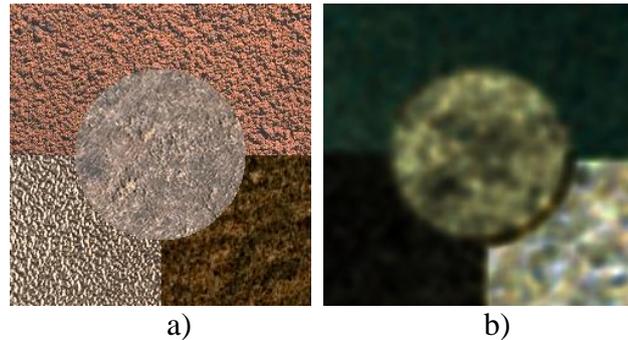


Fig. IV.5 : Une image couleur texture synthétique et son image  $T$   
a) L'image couleur synthétique b) Image  $T$

## 7 Résultats expérimentaux

Dans cette section, nous allons évaluer les performances de la segmentation des images couleur texturées en utilisant les champs aléatoire de Markov comme technique de segmentation et la classification spectrale pour l'extraction des paramètres de texture.

La segmentation markovienne décrite par l'algorithme ICM nécessite des valeurs initiales des vecteurs moyennes et des matrices de variances covariances entre les différentes composantes de chaque classe. Comme le but de ce travail est de démontrer l'efficacité des nouveaux paramètres proposés pour la séparation des différentes régions texturées, les résultats de la segmentation doivent être donc indépendants de l'étape d'initialisation. Nous proposons ainsi une classification initiale simple qui prend en compte les différentes classes couleur texture présentes dans l'image. Pour chaque classe, un certain nombre de pixels sont sélectionnés manuellement à partir de l'image affichée (voir Fig. IV.6). L'histogramme multidimensionnel de ces pixels est alors construit comme un ensemble d'hyper-cubes de dimension  $D$ , avec  $D$  correspondant au nombre de composantes de l'image. En supposant que chaque classe est représentée par un histogramme monomodal, le centre de la classe correspond alors à l'hyper-cube présentant un nombre maximum de pixels. Une fois les différents centres trouvés, chaque pixel de l'image est alors affecté à la classe qui présente le centre le plus proche de ce pixel au sens d'une distance euclidienne [16]. La figure IV.6 montre l'efficacité de cette technique de pré-segmentation comparativement à une pré-segmentation par l'algorithme EM.

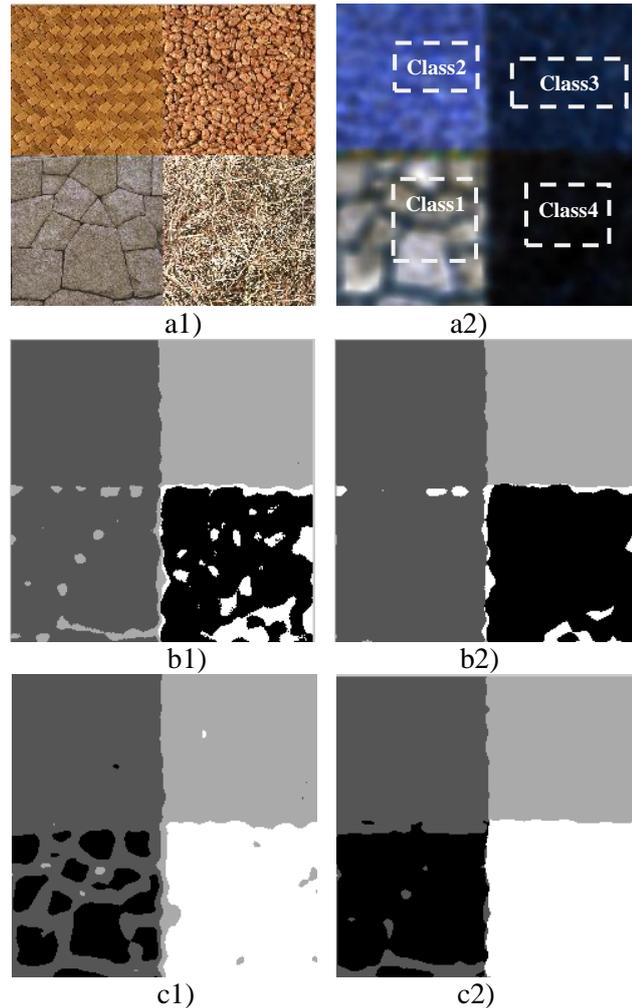


Fig. IV.6 : Résultats de la segmentation markovienne en utilisant les paramètres RGB-texture avec une pré-segmentation par EM et par la méthode proposée  
 a1) image couleur initiale    a2) Image de texture  
 b1) EM pré-segmentation    b2) Résultat de la segmentation avec une pré-segmentation EM  
 c1) pré-segmentation proposée    c2) Résultat de la segmentation avec la pré-segmentation proposée

Des images synthétiques (Fig. IV.7.a) et réelles (Fig. IV.8.a) sont utilisés afin d'effectuer les différentes expériences. Toutes ces images sont des images de couleurs naturelles et contiennent des régions texturées. Les images synthétiques sont créées à partir de la base de données Vistex [139]. Les images réelles proviennent de la base de données Berkeley [140].

L'utilisation d'images synthétiques permet de mesurer les performances de la segmentation par un taux de mal-classification, estimé par le quotient du nombre de pixels mal classés sur le nombre total des pixels dans l'image. Plus ce taux est faible, meilleures sont les résultats.

Toutes les expériences sont réalisées avec un large voisinage de taille (15x15) et les patches de taille (7x7). Le choix de ces tailles est réalisé de manière empirique.

En premier lieu, l'espace de couleur RGB est utilisé pour représenter les paramètres de couleur. Ensuite, pour étudier l'influence de l'espace colorimétrique, les résultats sont fournies pour les espaces colorimétriques CIE L\*a\*b\* et HSV.

Les images de la figure IV.7 et de la figure IV.8 sont organisées en colonnes. La première colonne a) contient les images initiales couleur texturées. La seconde b) affiche les images composées par les paramètres de texture extraits par la méthode proposée dans le paragraphe précédent. La troisième colonne c) fournies les résultats de segmentation en utilisant uniquement les paramètres RGB-couleur. La colonne d) montre les images segmentées en utilisant les paramètres RGB-texture. Dans la dernière colonne e), les images segmentées sont obtenues en combinant les paramètres RGB-couleur avec les paramètres RGB-texture.

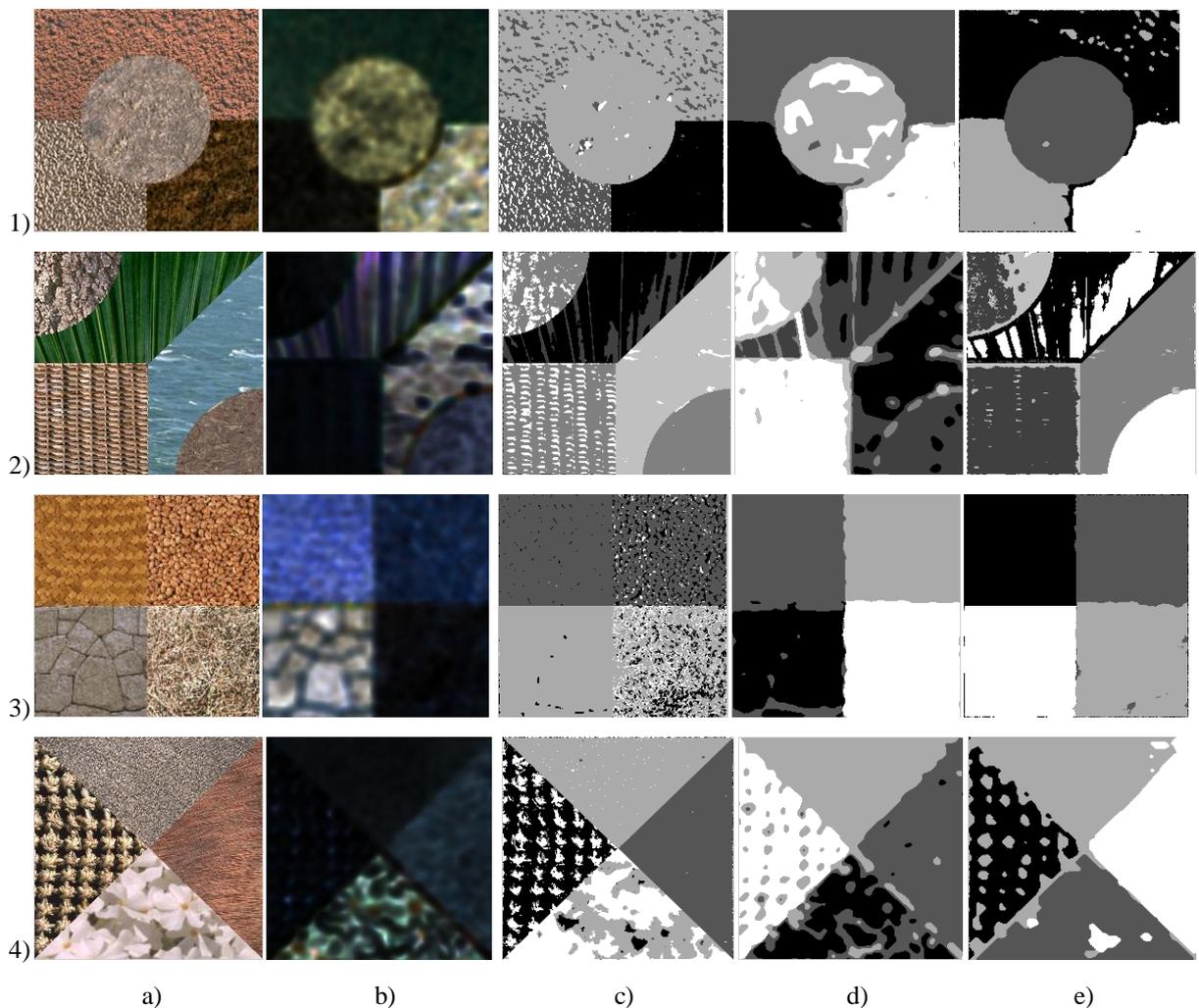


Fig. IV.7: Résultats de la segmentation des images synthétiques

- a) Image couleur b) Image des paramètres RGB-texture c) Résultats de la segmentation en utilisant seulement les paramètres RGB-couleur d) Résultats de la segmentation en utilisant seulement les paramètres RGB-texture e) Résultats de la segmentation en combinant les paramètres RGB-couleur et RGB-texture

Nous pouvons observer sur la figure IV.7 que dans tous les tests, la segmentation basée sur la combinaison des paramètres couleurs et textures fournit de bien meilleurs résultats que celles utilisant seulement les paramètres couleur ou seulement les paramètres texture. En effet, par exemple, la première ligne de la figure IV.7 montre que l'utilisation des paramètres RGB-couleur seulement (Fig. IV.7.1.c) produit une mauvaise segmentation car une seule texture parmi quatre (représentée en noir) est parfaitement détectée. Le résultat devient meilleur avec les paramètres de texture seulement (Fig. IV.7.1.d) puisque trois textures parmi quatre sont bien segmentées, seule la région représentée sous forme d'un cercle est mal détectée. La segmentation est bien meilleure lorsque les paramètres de couleur et de texture sont combinés (Fig. IV.7.1.e) parce que toutes les quatre textures sont bien séparées. Cette évaluation visuelle est confirmée par les taux d'erreur présentés dans le tableau IV.1.

Tableau IV.1: Taux de pixels mal classés sur les résultats de la segmentation utilisant les paramètres RGB-couleur seulement, les paramètres RGB-texture seulement ou combinant les paramètres RGB-couleur et RGB-texture

|                         | Utilisation des paramètres RGB-couleur | Utilisation des paramètres RGB-texture | Combinaison des paramètres |
|-------------------------|--|--|----------------------------|
| Image de la Fig. IV.7.1 | 46.75%                                 | 10.95%                                 | 1.24%                      |
| Image de la Fig. IV.7.2 | 39.77%                                 | 37.98%                                 | 18.24%                     |
| Image de la Fig. IV.7.3 | 40.72%                                 | 5.07%                                  | 5.08%                      |
| Image de la Fig. IV.7.4 | 19.20%                                 | 17.09%                                 | 11.85%                     |

Les résultats de segmentation de toutes les images synthétiques de la figure IV.7 sont les plus mauvais lorsque les paramètres RGB-couleur sont utilisés parce que, comme il est possible de l'observer sur le tableau IV.1, le taux de pixels mal classés est le plus élevé dans ce cas. L'utilisation des paramètres de texture seulement améliore les résultats. Ces améliorations peuvent être conséquentes, comme pour le cas de l'image de la figure IV.7.3 (voir tableau IV.1 ligne 3) ou non conséquente, comme pour le cas de l'image de la figure IV.7.2 (voir tableau IV.1 ligne 2). Nous supposons que cette amélioration n'est pas conséquente lorsque les textures présentes dans l'image sont hétérogènes ou entachées de bruits. Dans tous les cas, l'amélioration est conséquente lorsque les paramètres couleur et les paramètres texture sont combinés (voir la dernière colonne du tableau IV.1).

En plus des images de synthèse, quelques résultats sont fournis dans la figure IV.8 sur des images couleurs réelles extraites de la base de données Berkeley [140].

La figure IV.8.1 illustre que les résultats obtenus en utilisant les paramètres de texture seulement (Fig. IV.8.1.d) sont meilleurs que ceux obtenus en utilisant les paramètres couleur

uniquement (Fig. IV.8.1.c). Dans quelques images, les contours obtenus en utilisant les paramètres de texture sont flous (comme dans la figure IV.8.2.d) car les frontières des objets sont sombres dans l'image de texture extraite (Fig. IV.8.2.b). Ce problème se trouve résolu lorsque les paramètres couleur sont combinés aux paramètres de texture (Fig. IV.8.2.e).

Pour toutes les images testées, les résultats de la segmentation sont meilleurs lorsque les paramètres couleur et texture sont combinés car les contours sont plus nettes et les régions sont plus homogènes.

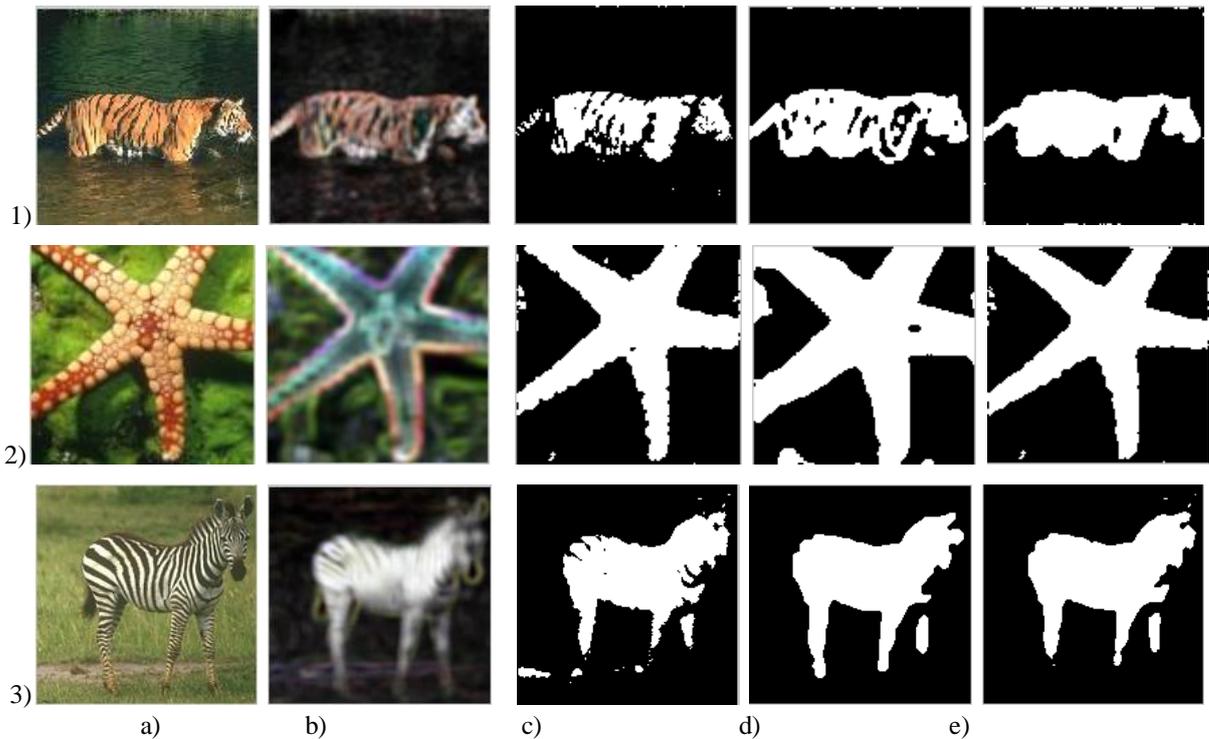


Fig. IV.8 : Résultats de la segmentation des images naturelles

a) L'image couleur b) Image des paramètres RGB-Texture c) Résultats de la segmentation en utilisant seulement les paramètres RGB-couleur d) Résultats de la segmentation en utilisant seulement les paramètres RGB-texture e) Résultats de la segmentation en combinant les paramètres RGB-couleur et RGB-texture

## 8 Influence de l'espace de représentation couleur

Pour examiner l'influence de l'espace de représentation colorimétrique, les mêmes images synthétiques de la figure IV.7 sont testées. Les espaces CIE  $L^*a^*b^*$  et HSV sont choisis pour réaliser ces tests. Les paramètres de texture sont alors extraits respectivement des composantes  $L$ ,  $a^*$  et  $b^*$  et à partir des composantes  $H$ ,  $S$  et  $V$ . La figure IV.9 illustre les résultats de la segmentation des images testées en utilisant les paramètres couleur et les paramètres texture extraits respectivement des trois espaces de représentation RGB,  $L^*a^*b^*$  et HSV.

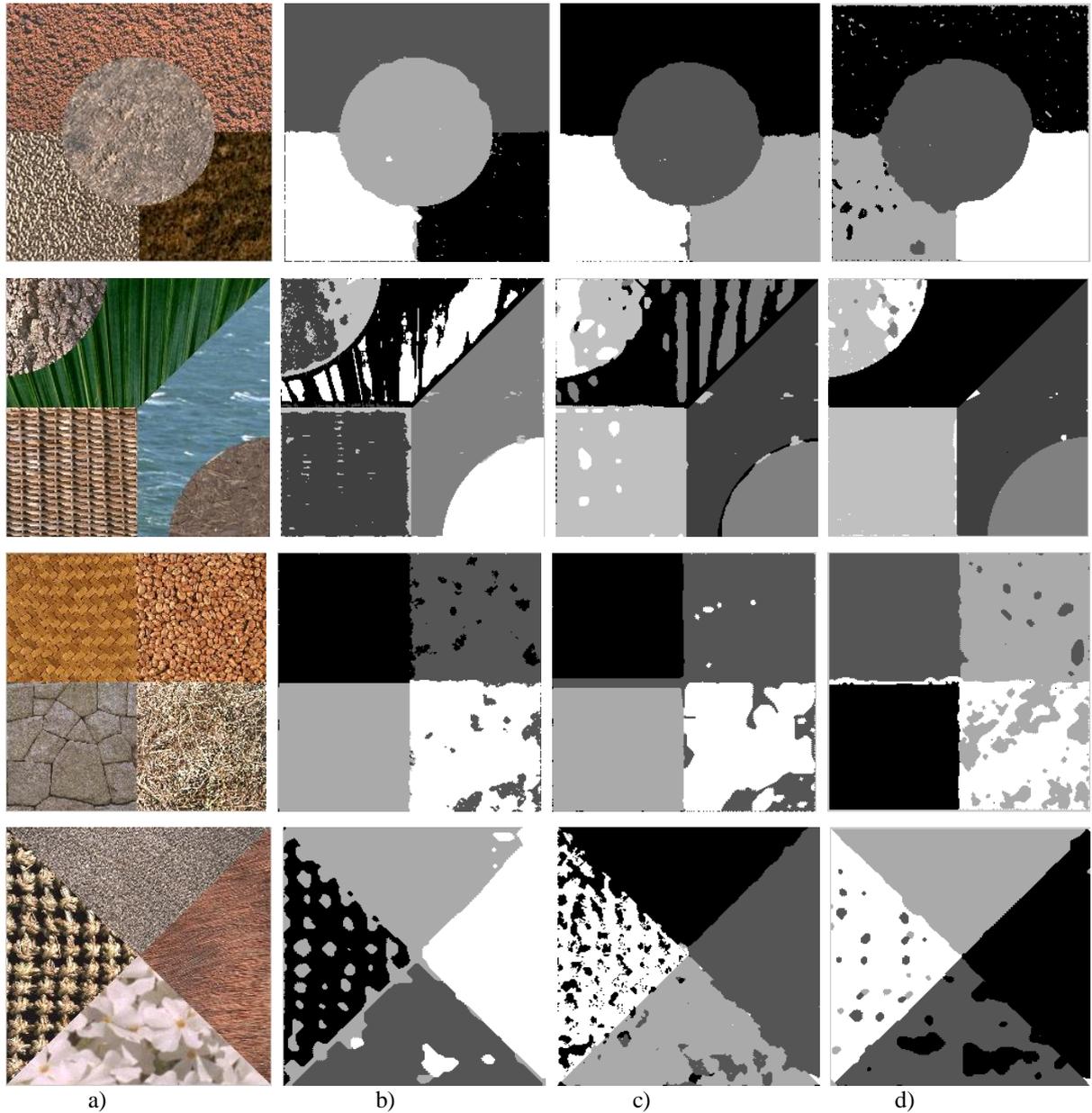


Fig. IV.9 : Résultats de la segmentation des images synthétiques en utilisant différents espace colorimétrique

- a) Image couleur originale
- b) Résultats de la segmentation en combinant les paramètres RGB-couleur et RGB-texture
- c) Résultats de la segmentation en combinant les paramètres  $L^*a^*b^*$ -couleur et  $L^*a^*b^*$ -texture
- d) Résultats de la segmentation en combinant les paramètres HSV-couleur et HSV-texture

Pour avoir une meilleure évaluation des différents résultats, les tableaux IV.2 et IV.3, fournissent le taux de pixels mal classés pour chaque résultat de la segmentation en utilisant respectivement les espaces CIE  $L^*a^*b^*$  et HSV.

Tableau IV.2: Taux de pixels mal classés sur les résultats de la segmentation utilisant les paramètres L\*a\*b\*-couleur seulement, les paramètres L\*a\*b\*-texture seulement ou combinant les paramètres L\*a\*b\*-couleur et L\*a\*b\*-texture

|                            | Utilisation des paramètres L*a*b*-couleur | Utilisation des paramètres L*a*b*-texture | Combinaison des paramètres (Fig.8.c) |
|----------------------------|---|---|--------------------------------------|
| Image original Fig. IV.7.1 | 41.44%                                    | 3.73%                                     | 3.22%                                |
| Image original Fig. IV.7.2 | 31.90%                                    | 21.09%                                    | 18.00%                               |
| Image original Fig. IV.7.3 | 44.02%                                    | 8.05%                                     | 7.39%                                |
| Image original Fig. IV.7.4 | 19.31%                                    | 12.14%                                    | 11.94%                               |

Tableau IV.3: Taux de pixels mal classés sur les résultats de la segmentation utilisant les paramètres HSV-couleur seulement, les paramètres HSV-texture seulement ou combinant les paramètres HSV-couleur et HSV-texture

|                            | Utilisation des paramètres L*a*b*-couleur | Utilisation des paramètres L*a*b*-texture | Combinaison des paramètres (Fig.8.d) |
|----------------------------|---|---|--------------------------------------|
| Image original Fig. IV.7.1 | 39.17%                                    | 26.64%                                    | 4.21%                                |
| Image original Fig. IV.7.2 | 39.34%                                    | 26.75%                                    | 9.19%                                |
| Image original Fig. IV.7.3 | 45.07%                                    | 14.05%                                    | 10.49%                               |
| Image original Fig. IV.7.4 | 39.67%                                    | 10.87%                                    | 8.00%                                |

Les résultats présentés dans les tableaux IV.1, IV.2 et IV.3 et dans la figure IV.9 montrent qu'aucun de ces trois espaces colorimétriques testés ne s'est avéré suffisamment supérieur. En effet, si le résultat est meilleur pour une image dans un espace de représentation spécifique, il n'est pas automatiquement meilleur pour une autre image dans cet espace. Ce résultat concorde avec les travaux de Drimbareau et Whelan [103] et de Sharbek et Koschan [104] qui ont étudié l'effet de l'utilisation de différents espaces couleur sur les résultats de leurs analyses.

La conclusion tirée précédemment sur l'espace de représentation RGB reste valable pour les deux autres espaces testés. Les nouveaux paramètres de texture que nous avons proposés se sont avérés efficaces car ils améliorent les résultats de la segmentation comparativement à l'utilisation des paramètres couleur et les résultats sont nettement meilleurs en combinant les paramètres couleur avec les paramètres texture dans chacun des trois espaces colorimétrique testés.

## 9 Conclusion

La couleur et la texture sont deux éléments importants dans la perception visuelle humaine. Plusieurs approches de segmentation ont utilisé ces deux éléments pour obtenir des régions homogènes. Dans cette étude, nous avons proposé un algorithme de segmentation

d'images couleur texturées qui combine les paramètres couleur et texture. Notre approche est différente des travaux antérieurs essentiellement dans la méthode utilisée pour l'extraction des paramètres de texture.

Les paramètres de texture proposés dans ce travail sont extraits des matrices de "marche aléatoire sur graphe" qui sont construites dans un large voisinage de chaque pixel prenant ainsi en compte la mesure de similarité entre les pixels de son voisinage. Ensuite, la segmentation est obtenue en classant les pixels dans différentes classes en utilisant le MAP (Maximum A Posteriori Probability). Ces classes sont représentées par des distributions gaussiennes multi-composantes dans le schéma d'une segmentation MRF. L'optimisation est réalisée par une technique itérative déterministe ICM avec une pré-segmentation proposée simple et prenant en compte les différentes classes de l'image.

Pour rendre compte de la contribution des paramètres de texture proposés, nous avons fournis trois types de résultats. Le premier consiste en la segmentation d'images en utilisant les paramètres couleur seulement, le second traite de la segmentation d'image en utilisant les paramètres texture seulement, et le dernier résultat consiste en la segmentation en combinant les paramètres couleur et texture.

Il est évident que la contribution des informations de texture "spectrales" est déterminante car elle améliore nettement les résultats. L'utilisation des paramètres de texture peut être suffisante dans certains exemples, mais les meilleurs résultats sont observés lorsque nous combinons les deux types de paramètres (couleur et texture).

Lorsque nous combinons les paramètres de texture avec les paramètres couleur, leurs avantages se complètent. Les paramètres couleur permettent une meilleure localisation des frontières tandis que les paramètres de texture fournissent une classification moins bruitée des régions.

## Conclusion générale

L'imagerie multi-composante est un domaine relativement récent qui est en pleine expansion. Cette thèse se veut être une contribution dans ce domaine et porte sur le traitement des images multi-composantes caractérisées par le fait que la valeur d'un pixel n'est plus représentable par un scalaire, mais par plusieurs composantes, qui peuvent être de même nature, ou bien correspondre à des grandeurs physiques différentes. Elle adapte différents algorithmes de classification pour segmenter des images multi-composantes. C'est ainsi que les algorithmes de classification EM, K-means, FCM et SOM ont été utilisés pour segmenter des images couleur et multi-spectrales. La mise en œuvre de ces techniques se ramène à un problème de combinaison des informations apportées par les différentes composantes qui peut s'effectuer à différents niveaux, sur les attributs radiométriques des pixels ou sur les primitives de segmentations marginales. En ce sens, nous avons proposé une nouvelle technique de combinaison des cartes des étiquettes en segmentation marginale et nous avons confronté les résultats obtenus aux résultats de la segmentation vectorielle via une évaluation par la fonction de Borsotti. Cette technique s'est avérée appréciable surtout dans le cas où les résultats de la segmentation vectorielle fournissent des cartes des étiquettes présentant un nombre important de régions de faible dimension. En effet, ces petites régions qui sont éliminées dans la stratégie de segmentation marginale développée, pénalisent la fonction de Borsotti dans la segmentation vectorielle.

Nous avons ensuite utilisé une approche markovienne pour segmenter des images de télédétection et des images couleur texturées. Les champs de Markov sont très intéressants dans le sens où ils ne se limitent pas à la prise en compte individuelle des pixels mais ils permettent de prendre en compte les relations de voisinage entre les pixels. Ainsi, la probabilité qu'un pixel appartienne à une classe dépend non seulement de son "intensité", mais aussi de celles de ses voisins. Il est ainsi rare d'avoir des régions de très faible dimension. Dans ce formalisme, le champ des étiquettes est modélisé par une énergie de

Gibbs. Plusieurs formes d'énergie de Gibbs, dont deux que nous avons nouvellement proposées, ont été testées et comparées dans le cadre de la segmentation des images MSG2. Cette étude comparative n'est pas totalement exhaustive car le nombre de possibilités pour le choix de celle-ci est très important et il aurait été impossible de les examiner toutes. Seules les plus connues ou celles qui n'ont pas été appliquées à la segmentation des images satellitaires ont été prise en considération. Plusieurs formes de ces énergies ont fournies des résultats appréciables, y compris les deux que nous avons proposées

La couleur et la texture sont des caractéristiques largement observées dans la plupart des images. De nombreuses approches pour l'extraction des paramètres de texture ont été développées. Dans ce travail, nous avons proposé de nouveaux paramètres de texture basée sur les marches aléatoires sur graphes (random walk in graph) qui est déduite du domaine de la classification spectrale. Pour rendre compte de la contribution de ces paramètres proposés, nous avons fourni trois types de résultats. Le premier consiste en la segmentation d'images en utilisant les paramètres couleur seulement, le second traite de la segmentation d'image en utilisant les paramètres texture seulement, et le dernier résultat consiste en la segmentation en combinant les paramètres couleur et texture. Il a été constaté que la contribution des informations de texture "spectrales" est déterminante car elle améliore nettement les résultats.

Les tests réalisés dans ce travail portent sur deux domaines d'imagerie en plein essor, à savoir l'imagerie couleur et multi-spectrale. A travers les traitements réalisés sur ces images, nous avons principalement étudié deux types d'images multi-composantes, classées selon la nature des composantes:

- les images quasi-homogènes, dont les composantes mesurent la même grandeur, sans être exactement similaires tel que les images couleurs dans la représentation RGB et les images Meteosat multi-spectrale;
- les images hétérogènes, caractérisées par des composantes de natures différentes tel que les images couleurs dans les représentations CIELab et HSV et les images couleur combinant les informations couleur et texture.

Les perspectives de cette thèse sont nombreuses.

Dans le cas de la segmentation marginale des images multi-composantes, il serait intéressant d'étudier des situations où le nombre de composantes est important. Il sera probablement nécessaire de modifier les stratégies de combinaison des résultats fournis par les différentes composantes, pour des raisons de rapidité dans les calculs bien sûr, mais aussi pour une meilleure prise en compte de l'information utile. La méthode de fusion proposée commence par fusionner les images des étiquettes en attribuant un numéro différent pour

chaque combinaison d'étiquettes des images mono-composantes. Ce nombre peut être important et entraîner une image de fusion dont le nombre de classes de faible taille est très important. Il est évident que l'élimination des régions et des classes de petites tailles proposées peut être une solution à ce problème sauf que, cela risque de prendre un temps considérable. Il est alors nécessaire de réfléchir à une méthode qui permettra d'incorporer ces régions et classes à celles dont la taille est plus importante, durant la technique de fusion.

Dans la méthode de segmentation d'images basée sur le formalisme markovien, en plus de la modélisation du processus des étiquettes par une fonction de Gibbs, le processus image est modélisé par une distribution gaussienne multidimensionnelle, or l'information du contexte spatial peut améliorer la classification lorsque la forme de la loi conditionnelle est bien adaptée. Il serait ainsi intéressant d'exploiter des lois autres que la loi gaussienne dans la modélisation de la probabilité conditionnelle.

Dans ce travail, nous avons aussi proposé de nouveaux paramètres de texture que nous avons utilisés pour réaliser la segmentation des images couleur texturées. Nous préconisons d'utiliser ces nouveaux paramètres pour caractériser des images entières en vue de leurs classification.

## Références bibliographiques

- [1] G. P. Dinneen, Programming Pattern Recognition, Proceedings Western Joint Computer Conference, March 1955, pp. 94-100.
- [2] Grant B. Anderson, Thomas S. Huang, Frequency-domain image errors, Pattern Recognition, vol. 3, 1961, pp. 185-196.
- [3] R. E. Graham, Snow-Removal: A Noise-Stripping Process for Picture Signals, IRE Trans. Inf. Theory, IT-8, vol. 1, 1962, pp. 129-144.
- [4] Patrick Lambert, Etudes méthodologiques du filtrage et de la segmentation d'images multicomposantes. Habilitation à Diriger des Recherches HDR, Université de Savoie, France 2002.
- [5] Sié Ouattara, Stratégies de segmentation d'images multicomposantes par analyse d'histogrammes multidimensionnels : application à des images couleur de coupes histologiques de pommes, thèse de doctorat en Traitement du signal et des images, Ecole doctorale STIM (Sciences et Technologies de l'Information et de Mathématiques), Angers, 2009.
- [6] <http://www.eumetsat.int>
- [7] Peter Yu, Segmentation of RADARSAT-2 Dual-Polarization Sea Ice Imagery, A thesis of Waterloo University, Ontario, Canada, for the degree of Master of Applied Science in Systems Design Engineering, 2009
- [8] Gabriel VASILE, Imagerie Radar à Synthèse d'Ouverture interférométrique et polarimétrique, Application au suivi des glaciers alpins, thèse de Doctorat, présentée conjointement à l'université de Savoie, et à l'université Polytechnique de Bucarest, 2007
- [9] T. Young, On the theory of light and colors, Philosophical Transactions of the Royal Society, vol. 91, 1802, p. 12-49.
- [10] H. Grassmann, On the theory of compound colours, Ann. Phys. Leipzig, vol.9, 1854 (traduction anglaise)
- [11] J. Maxwell, Experiments of colour, as perceived by the eye, with remarks on colour blindness, Transactions of the Royal Society of Edinburg, vol. 21, 1857
- [12] <http://www.itres.com>
- [13] <http://makalu.jpl.nasa.gov/html/aboutav.html>
- [14] K. Tang, M. Gabbouj, J. Astola, A Novel Nonlinear Vector Filter for Grey-Level Image Filtering, Proc. IEEE International Workshop on Non Linear Signal and Image Processing, Neos Marmaras, Haldikiki, Greece, June 1995, pp. 626-629.

- [15] Ph. Bolon, R. Kara Falah, Filtering in the segmentation space, Proc. of the IEEE Conf. on Acoustics Speech and Signal Processing, ICASSP'97, Munich, Germany, April 1997, pp. 3161-3164.
- [16] Sadia Alkama, Youssef Chahir, Daoud Berkani, Color Textured Image Segmentation: An Approach Combining Random Walks on Graph and Markov Random Fields, submitted to Pattern Analysis and Applications
- [17] A. Trémeau, C. Fernandez-Maloigne, P. Bonton, Image numérique couleur : de l'acquisition au traitement, Editions Dunod, Paris, France, 2004.
- [18] Robert Sève, Physique de la couleur : de l'apparence colorée à la technique colorimétrique, Physique fondamentale et appliquée, Editions Masson, 1<sup>ère</sup> édition, Paris, France, 1996
- [19] P. Kowaliski, Vision et mesure de la couleur, Physique fondamentale et appliquée, Editions Masson, 2<sup>ème</sup> édition, Paris, France, 1990
- [20] J. Guild, A critical survey of modern developments in the theory and technique of colorimetry and allied sciences, Proc. Opt Convention, 1926, pp. 61-146
- [21] W. D. Wright, 50 years of the 1931 CIE Standard Observer for colorimetry, Farbe, 1<sup>st</sup> edition, 1981
- [22] N. Vandenbroucke, Segmentation d'images couleur par classification de pixels dans des espaces d'attributs colorimétriques adaptés: Application à l'analyse d'images de football, Thèse de doctorat de l'Université de Lille 1, France, 2000
- [23] Commission Internationale de l'Eclairage, Colorimetry, Tech. Rep., Bureau central de la CIE, 1986
- [24] Konstantinos N. Plataniotis, Anastasios N. Venetsanopoulos, Color image processing and applications Springer, 2000.
- [25] A. Trémeau, Contribution des modèles de la perception visuelle à l'analyse d'images couleur, Thèse de doctorat de l'Université Jean Monnet, Saint-Etienne, France, 1993.
- [26] W. S. Stiles et J. M. Burch, NPL colour-matching investigation : final report, Optica Acta, vol. 6 1959, pp. 1-26.
- [27] A. Stockman, L.T. Sharpe, The spectral sensitivities of the middle and long Wavelength-sensitive cones derived from measurements in observers of known genotype, Vision Research, vol. 40, 2000, pp. 1711-1737.
- [28] Françoise Viénot, Colorimétrie et image couleur, Ecole de printemps-images couleurs, Pau, GdR-ISIS, 2001, pp. 2-3.
- [29] O. D. Faugeras, Digital color image processing within the framework of a human visual model, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 27, no 4, 1979, pp. 380-393.
- [30] Q.T. Luong, La couleur en vision par ordinateur : une revue, Traitement du Signal, vol. 8, no 1, 1991, pp. 3-34.
- [31] Y. Ohta, T. Kanade, T. Sakai, Color information for region segmentation, Computer Graphic and Image Processing, vol. 13, 1980, pp. 222-241.

- [32] E. Hering, *Outlines of a Theory of the Light Sense*, Harvard University Press, Cambridge, MA, 1964.
- [33] AFNOR, *Couleur colorimétrie*, Afnor, 1989
- [34] R.C. Gonzalez, Woods R.E. *Digital image processing*. Addison-Wesley, Reading, Massachusetts, USA, 1993
- [35] H.D. Cheng, X.H. Jiang, Y. Sun, J. Wang, Color image segmentation: advances and prospects. *Pattern Recognition*, vol. 34, 2001, p. 2259-2281.
- [36] Olivier Lezoray, *Segmentation d'images couleur par morphologie mathématique et classification de données par réseaux de neurones : application à la classification de cellules en cytologie de séreuses*. Thèse de doctorat de l'Université de Caen, France, 2000.
- [37] Olivier Lezoray, Christophe Charrier, *Segmentation d'images couleur par coalescence non supervisée d'histogrammes 2D et fusion de régions selon la théorie de Dempster-Shafer*, *Traitement du Signal*, vol. 21, numéro spécial "L'image numérique couleur", 2004, pp. 605-621.
- [38] Christophe Charrier, Olivier Lezoray, *Segmentation d'images couleur par la théorie de l'évidence*, CORESA (COMpression et REprésentation des Signaux Audiovisuels), 2005, pp. 291-296.
- [39] Radja Khedam, Abdenour Bouakache, Grégoire Mercier, Aichouche Belhadj-Aissa, *Fusion multirate à l'aide de la théorie de Dempster-Shafer pour la détection et la cartographie des changements : application aux milieux urbain et périurbain de la région d'Alger*, *Téledétection*, vol. 6, no 4, 2006, pp. 361-403
- [40] A.P. Dempster, *Upper and Lower Probabilities Induced by a Multi-valued Mapping*, *Annals of Mathematical Statistics*, vol. 38, 1967.
- [41] A.P. Dempster, *A Generalization of Bayesian Inference*, *Journal of the Royal Statistical Society*, vol. 30, série B, 1968 .
- [42] G . Shafer, *A Mathematical Theory Of Evidence*, Princeton University Press, Princeton, 1976.
- [43] A.-S. Capelle, O. Colot, and C. Fernandezmaloigne, *Evidential segmentation scheme of multi-echo mr images for the detection of brain tumors using neighborhood information*, *Information Fusion*, vol. 5, no 3, 2004, pp. 203-216.
- [44] F. Kurugollu, B. Sankur, A.E. Harmanci, *Color image segmentation using histogram multithresholding and fusion*, *Image and Vision Computing*, 19, 2001, pp. 915-928
- [45] Olivier Lézoray, Christophe Charrier, *Color image segmentation using morphological clustering and fusion with automatic scale selection*, *Pattern Recognition Letters*, vol. 30, 2009, pp. 397-406
- [46] M. A. Tanner, *Tools for statistical inference*. Springer-Verlag, 1996.
- [47] G. F. McLachlan, T. Krishnan, *The EM algorithm and Extensions*, John Wiley and Sons, 1997.
- [48] J. L. Schafer, *Analysis of Incomplete Multivariate Data*, Chapman and Hall/CRC, 1997.
- [49] Kamal Hammouche, *Méthodes de classification automatique utilisant la notion de texture*, Thèse de Doctorat d'Etat, Département d'Automatique, Faculté de Génie Electrique et Informatique, Université Mouloud Mammeri, Tizi-Ouzou, 2007

- [50] C. Wu, On the convergence properties of the EM algorithm, *Annals of Statistics*, vol. 11, 1983, pp. 95–103.
- [51] G. McLachlan, D. Peel, *Finite Mixture Models*, Wiley Interscience, New York, 2000.
- [52] P. Berkshire, *Survey of Clustering Data Mining Techniques*, 2002
- [53] V. Faber, *Clustering and Continuous K-means*, Los Alamos Science, 1994
- [54] J. MacQueen, Some Methods for Classification and Analysis of Multivariate Observations, *Proc. 5th Berkeley Symp.*, 1965, pp. 281-297.
- [55] E. R. Ruspini, A New Approach to Clustering, *Inform. Control*, vol. 15, no. 1, 1969, pp. 22-32.
- [56] E. R. Ruspini, Numerical Methods for Fuzzy Clustering, *Information Science*, vol. 2, 1970, pp. 319-350.
- [57] J. C. Dunn, A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact Well-Separated Clusters, *J. Cybernetics*, vol. 3, no. 3, 1973, pp. 32-57.
- [58] J. C. Bezdek, *Fuzzy mathematics in pattern classification*, PH.D dissertation, Cornell Univ., Ithaca, NY, 1973.
- [59] J. C. Bezdek, Numerical taxonomy with fuzzy sets, *Journal of Mathematical Biology*, 1, 1974, pp. 57-71.
- [60] T. Kohonen, *Self organization and associative memory*, Springer, Heidelberg, 1984
- [61] T. Kohonen, *Self Organizing Maps*, Springer, 3<sup>e</sup> edition, 2001
- [62] G. Dreyfus, J. M. Martinez, M. Samuelides, M. B. Gordon, F. Bardan, S. Thiria et L. Herault, *Réseaux de neurones : méthodologie et application*, Edition EYROLLES, 2004
- [63] A. K. Jain, R.C. Dubes, *Algorithms for clustering data*, Prentice Hall, 1988
- [64] Sadia Alkama, Youssef Chahir, Daoud Berkani , Segmentation des Images Couleur par la Carte Auto-Organisatrice de Kohonen et la Classification Ascendante hiérarchique, 2nd International conference on Systems and Information Processing, Guelma, Algeria, May 15-17, 2011
- [65] H. Zhang, J. E. Fritts, S. A. Goldman, Image segmentation evaluation: A survey of unsupervised methods, *Computer Vision and Image Understanding*, vol. 110, no 2, 2008, pp. 260-280
- [66] Y. J. Zhang, A survey of evaluation methods of image segmentation, *Pattern Recognition*, vol. 29, no 8, 1996, pp. 1335-1346
- [67] J. Liu, Y.-H. Yang, Multiresolution color image segmentation, *IEEE transaction on Pattern Analysis and Machine Intelligence*, vol. 16, no7, 1994, pp. 689-700
- [68] M. Borsotti, P. Campadelli, R. Schettini, Quantitative evaluation of color image segmentation results, *Pattern Recognition Letters*, vol. 19, 1998, pp.741-747
- [69] Hui Zhou, Runsheng Wang, Cheng Wang, A novel extended local-binary-pattern operator for texture analysis, *Information Sciences*, vol. 178, 2008, pp.4314-4325.
- [70] X. Mei, Qi. Wang, Qu. Wang, W. Lin, Research on remote sensing image segmentation based on ant colony algorithm- take the land cover classification of middle Qinling mountains for

- example, International Symposium on Multispectral image processing and pattern recognition N°6, Yichang Shi, China, 2009.
- [71] Q. Dai, G. Liu, C. Wang, L. Wang, A remote sensing image segmentation method based on spectral and structure information fusion, *The international archives of the photogrammetry, Remote sensing and spatial information sciences*, vol. 38, 2008, Beijing.
- [72] S. Zhanfeng, L. Jiancheng, H. Xiaodong, S. Weigang, A mean shift multi-scale segmentation for remote sensing images, *Geomatics inf. Sci. Wuhan Univ.*, vol.35, no3, 2010, pp. 313-316.
- [73] D. Li, G. Zhang, Z. Wu, L. Yi, An edge embedded marker-based watershed algorithm for high resolution remote sensing image segmentation, *IEEE transaction on image processing*, vol. 19, no 10, 2010, pp. 2781-2787.
- [74] L. Jiao, M. Gong, S. Wang, B. Hou, Z. Zheng, Q. Wu, Natural and remote sensing image segmentation using Memetic computing, *Computational intelligence Magazine IEEE*, vol. 5, no 2, 2010, pp. 78-91.
- [75] Huawu Deng, David A. Clausi, Unsupervised image segmentation using a simple MRF model with a new implementation scheme, *Pattern Recognition*, vol. 37, 2004, pp. 2323-2335.
- [76] J. Keuche, S. Naumann, M. Heiler, A. Siegmund, Automatic land cover analysis for Tenerife by supervised classification using remotely sensed data, *Remote Sensing of Environment*, vol. 86, no 4, 2003, pp. 530-541
- [77] G. Poggi, G. Scarpa, J.B. Zerubia, Supervised segmentation of remote sensing images based on a tree-structured MRF model, *IEEE Transaction on Geosciences and remote sensing*, vol. 43, no 8, 2005, pp. 1901-1911.
- [78] Brandt Tso and Richard C. Olsen, A contextual classification scheme based on MRF model with improved parameter estimation and multiscale fuzzy line process, *Remote Sensing of Environment*, vol. 97, no 1, 2005, pp. 127-136.
- [79] Sadia Alkama, Youssef Chahir, Daoud Berkani, Markovian approach using several Gibbs energy for remote sensing images segmentation, *Analog Integrated Circuits and Signal Processing*, 2011, vol. 69, pp. 39-47, DOI 10.1007/s10470-011-9631-8
- [80] Sadia Alkama, Daoud Berkani, Segmentation Mono et Multi Composante des Images Multi-Spectrales MSG, International Conference on Systems and Information Processing (ICSIP09), Guelma, Algeria, May 2-4, 2009.
- [81] Sadia Alkama, Daoud Berkani, Segmentation Vectorielle des Images Multi-spectrales MSG2 par une Approche Markovienne, *Studia Informatica Universalis*, Vol.8, no 4, 2010, pp. 89-110
- [82] J. E. Besag, Spatial interaction and the statistical analysis of lattice systems, *Journal of the Royal Statistical Society, series B*, vol. 36, 1974, pp. 192-326.
- [83] S. Geman, D. Geman, Stochastic relaxation, gibbs distribution, and the Bayesian restoration of images, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, n°6, 1984, pp. 721-741.
- [84] Sadia Alkama, Youssef Chahir et Daoud Berkani, Markovian Approach for Remote Sensing Images Segmentation, International Conference on Electrical Engineering, Electronics and Automatic'10 (ICEEEA'10), Bejaia, Algeria, 2-3 November 2010.
- [85] J. P. Cocquerez, S. Philipp, *Analyse d'images : filtrage et segmentation*, MASSON, Paris 1995

- [86] J. M. Hammersley and P. Clifford, Markov fields on finite graphs and lattices, unpublished, 1971.
- [87] J. Besag, On the Statistical Analysis of Dirty Pictures, J. Royal Statistical Soc., Series B, vol. 48, 1986, pp. 259-302.
- [88] Dileep Kumar Panjwani, Glenn Healey, Markov random field models for unsupervised segmentation of textured color images. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.7, no. 10, 1995, pp. 939-954.
- [89] Fabien Salzenstein and Wojciech Pieczynski, Parameter Estimation in Hidden Fuzzy Markov Random Fields and Image Segmentation, Graphical Models and Image Processing, vol. 59, no 4, 1997, pp. 205–220.
- [90] Sadia Alkama, Daoud Berkani et J. C. Berges, Comparaison de plusieurs formes d'énergies de Gibbs dans l'approche Markovienne de la segmentation des images MSG, 2<sup>ième</sup> Colloque de Recherche Appliquée et de Transfert de Technologie 1<sup>ère</sup> Conférence Internationale CRATT'2007, ISET Rades, Tunisie, 14 et 15 Novembre 2007
- [91] R. Lamotte, R. Alt, Comparison of simulated annealing algorithm for image restoration, Mathematics and computers in simulation, vol. 37, 1994, pp. 1-15
- [92] L. Blanc-Feraud, M. Barlaud, Restauration d'images bruitée par analyse multi résolution et champs de Markov, Traitement du signal, Vol. 10, N°.2, 1990, pp. 171-184.
- [93] T. Pappas, An adaptive clustering for image segmentation, IEEE Transaction on signal processing, vol. 40, no 4, 1992, pp. 901-914.
- [94] I. Jobard, M. Desbois, Remote sensing of rainfall over tropical Africa using Meteosat infrared imagery: sensitivity to time and space averaging, Int. J. Remote Sensing, vol. 13, no 14, 1992, pp. 2683-2700.
- [95] K. Ramage, Contribution à l'estimation des précipitations et à la caractérisation des nuages en Afrique de l'Ouest à partir des observations des satellites METEOSAT et TRMM, Thèse de doctorat de l'université PARIS 7, 2002
- [96] A.F. Hasler, J. Stong, R.H. Woodward, H. Pierce, Automatic analysis of stereoscopic satellite image pairs for determination of cloud-top height and structure, J. Appl. Meteor. vol. 30, 1991, pp. 257-281
- [97] S.J. Nieman, J. Schmetz, W.P. Menzel, , A comparison of several techniques to assign heights to cloud tracers, J. Appl. Meteor., vol. 32, 1993, pp.1559-1568
- [98] Zoltan Kato, Ting-Chuen Pong, A markov random field image segmentation model for color textured images, Image and vision computing, vol. 24, 2006, pp. 1103-1114.
- [99] Sadia Alkama, Daoud Berkani et J. C. Berges, Segmentation Markovienne des images MSG, 3<sup>ème</sup> Journées Internationales sur l'Informatique Graphique JIG2007, Constantine, Algérie, 29 et 30 Octobre 2007
- [100] Sadia Alkama, Kamal Hammouche, Segmentation d'Images Couleur par les Champs de Markov, Conférence sur la Vision Artificielle CVA'2007, Tizi Ouzou, Algérie, 18-20 Novembre 2007.
- [101] Kie B. Eom, Segmentation of monochrome and color textures using average modeling approach, Image and Vision Computing, vol. 17, 1999, pp. 233-244.

- [102] Mihran Tuceryan, Anil K. Jain, Texture analysis, Chapter 2, The Handbook of Pattern Recognition and Computer Vision (2<sup>nd</sup> edition), by C. H. Chen, L. F. Pau, P. S. P. Wang(eds.), World Scientific Publishing Co., 1998, pp. 207-248.
- [103] Alexandru Drimbareau and Paul F. Whelan Experiments in colour texture analysis, Pattern Recognition Letters, vol. 22, 2001, pp. 1161-1167.
- [104] Yining Deng, B. S. Manjunath, Unsupervised segmentation of color-texture regions in images and video, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no 8, 2001, pp. 800-810.
- [105] Michal Haindl, Stanislav Mikres, Colour texture segmentation using modelling approach, International Conference on Advances in Pattern Recognition ICAPR 2005, Lecture Notes in Computer Science, vol. 3687, 2005, pp. 484-491.
- [106] Zoltan Kato, Ting-Chuen Pong and Song Guo Qiang, Unsupervised segmentation of color textured images using a multi-layer MRF model, in proceedings of International Conference on Image Processing, vol I, IEEE, Barcelona, Spain 2003, pp. 961-964.
- [107] Mohand Saïd Allili, Djamel Ziou, Globally adaptive region information for automatic color-texture image segmentation, Pattern recognition Letters, vol. 28, 2007, pp. 1946-1956.
- [108] Lilong Shi, Brian Funt, Quaternion color texture segmentation, Computer Vision and Image Understanding, vol. 107, 2007, pp. 88-96.
- [109] Mofakharul Islam, John Yearwood, Peter Vamplew, Unsupervised color textured image segmentation using Cluster Ensembles and MRF model, Advances in Computer and Information Sciences and Engineering, 2008, pp. 323-329.
- [110] Jong-Sung Kim, Ki-Sang Hong, Color-texture segmentation using unsupervised graph cuts, Pattern Recognition, vol. 42, 2009, pp.735-750.
- [111] Thomas Brox, Mikaël Rousson, Rachid Derich, Joachim Weickert, Colour, texture, and motion in level set based segmentation and tracking, Image and Vision Computing, vol. 28, 2010, pp. 376-390.
- [112] W. E. Donath, A. J. Hoffman, Lower bands for the partitioning of graphs, IBM Journal of Research and Development, vol. 17, 1973, pp. 420-425.
- [113] Miroslav Fiedler, Algebraic connectivity of graphs, Czechoslovak Mathematical Journal, vol. 23, 1973, pp.298-305.
- [114] M. Bolla, Relation between spectral and classification proprieties of multigraphs, Technical Report No. DIMACS-91-27, Center for Discrete Mathematics and Theoretical Computer Science, 1991.
- [115] Bruce Hendrickson, Robert Leland, An improved spectral graph partitioning algorithm for mapping parallel computation, SIAM Journal on Scientific Computing, vol. 16, 1995, pp. 452-469.
- [116] Ulrike Von Luxburg, A tutorial on spectral clustering, Statistics and Computing, vol. 17, 2007, pp. 395-416.
- [117] Maurizio Filippone, Francesco Camastra, Francesco Masulli, Stefano Rovetta, A survey of kernel and spectral methods for clustering Pattern Recognition, vol. 41, 2008, pp. 176-190.

- [118] L. Zelnik-Manor, P. Perona, Self-tuning spectral clustering, *Advances in Neural Information Processing Systems*, vol. 17, 2004, pp. 1601-1608.
- [119] F. Chung, *Spectral graph theory*, volume 92 of *CBMS Regional Conference Series in Mathematics*, Published for the Conference Board of the Mathematical Sciences, Washington, DC, 1997.
- [120] B. Mohar, The Laplacian spectrum of graphs, In *Graph theory, combinatorics, and applications*. vol. 2 (Kalamazoo, MI, 1988), Wiley-Intersci. Publ., Wiley, New York, 1991, pp. 871-898.
- [121] B. Mohar, Some applications of Laplace eigenvalues of graphs, In G. Hahn and G. Sabidussi (Eds.), *Graph Symmetry: Algebraic Methods and Applications* (Vol. NATO ASI Ser. C 497, Kluwer, 1997, pp. 225-275
- [122] J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no 8, 2000, pp. 888-905.
- [123] A. Ng, M. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems 14*, MIT Press, 2002, pp. 849-856.
- [124] George E.P. Box, Gwilyn M. Jenkins, Gregory C. Reinsel, *Time series analysis: Forecasting and control*, 3<sup>rd</sup> ed., Prentice Hall, 1994.
- [125] Charles Kervrann, Fabrice Heitz, A Markov random field model based approach to unsupervised texture segmentation using local and global spatial statistics, *IEEE Trans. on Image Processing*, vol. 4, no 6, 1995, 856-862.
- [126] Santhana Krishnamachari, Rama Chellappa, Multiresolution Gauss-Markov random field models for texture segmentation, *IEEE Transactions on Image processing*, vol. 6, no 2, 1997, pp. 251-267.
- [127] Jian-Lin Chen, Amlan Kundu, Rotation and gray scale transform invariant texture identification using wavelet decomposition and hidden Markov model, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no 2, 1994, pp.208-214.
- [128] Kouroush Jafari-Khouzani, Hamid Soltanian-Zadeh, Radon transform orientation estimation for rotation invariant texture analysis, *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. 27, no 6, 2005, pp.1004-1008.
- [129] B.B. Chaudhuri, Nirupam Sarkar, Texture segmentation using fractal dimension, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 17, no 1, 1995, pp. 72-77.
- [130] Robert M. Haralick, Linda G. Shapiro, *Computer and robot vision vol.1*, Addison Wesley, 1992.
- [131] Ashok Samal, James R. Brandle, Dong Sheng Zhang, Texture as the basis of individual tree of identification, *Information Sciences*, vol. 176, 2006, pp.565-576.
- [132] D. Dunn, W.E. Higin, J. Wakely, Texture segmentation using 2-D Gabor elementary functions; *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no 2, 1994, pp.130-149.
- [133] Frank Y. Shih, Chao-Fa Chuang, Automatic extraction of head and face boundaries and facial features, *Information Sciences*, vol. 158, 2004, pp.117-130.
- [134] Alireza Tavakoli Targhi, Eric Hayman, Jan-Olof Eklundh, Mehrdad Shahshahani, The eigen-transform and Application, *Lecture Notes in Computer Science*, vol. 3851, 2006, pp. 70-79

- [135] L. Lovasz, Random walks on graphs: a survey, In *Combinatorics, Paul Erdos is eighty*, Budapest: Janos Bolyai Math. Soc., 1993, pp. 353–397.
- [136] B. Nadler, S. Lafon, R.R. Coifman, I.G. Kevrekidis, Diffusion maps, Spectral clustering and eigenfunctions of Fokker-Planck operators, In Y. Weiss, B. Schölkopf, J. Platt (Eds.), *Advances in Neural Information Processing Systems*, vol. 18, 2006, pp. 955-962, Cambridge, MA:MIT Press.
- [137] David Aldous, James Fill, *Reversible Markov Chains and Random Walks on Graphs*, University of California Berkeley 2002, online version available at <http://www.stat.berkeley.edu/users/aldous/RWG/book.html>.
- [138] R.R. Coifman, S. Lafon, A.B. Lee, M. Maggioni, B. Nadler, F. Warner, S.W. Zucker Geometric diffusion as tool for harmonic analysis and structure definition of data, Parts I and II, *Proceeding of the National Academy of Sciences*, vol. 102, no 21, 2005, pp. 7426-7437.
- [139] MIT VisTex, texture database, 1995 (<http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html>).
- [140] Berkeley Segmentation Dataset (URL <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html>).
- [141] Wladyslaw Skarbek and Andreas Koschan, *Colour image segmentation – A survey*, Technical report 93-32, Technical University Berlin, Oct 1994.